Short-term Rainfall Forecasting using Artificial Neural Networks

by

Vijay Misra

DEPARTMENT OF CIVIL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

December, 1999

Short-term Rainfall Forecasting using Artificial Neural Networks

A Thesis Submitted
in Partial Fulfillment of the Requirements
for the Degree of
Master of Technology

by
Vijay Misra

to the
Department of Civil Engineering
Indian Institute of Technology, Kanpur
December 1999

Certificate

Certified that the work contained in the thesis entitled "*Short-term Rainfall Forecasting using Artificial Neural Networks*", by Mr. *Vijay Misra*, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

(Dr. Ashu Jain)
Assistant Professor,
Department of Civil Engineering,
Indian Institute of Technology,
Kanpur.

December 1999

Abstract

Artificial Neural Network (ANN) models alongwith three other models namely Regression models, Thunderstorm cell models and Persistence model, using inputs as temperature, dewpoint temperature and relative humidity, have been developed for an hourly rainfall forecast for the Indianapolis Eaglecreek airport, Indiana state, USA. Forecasts from the year 1997 were verified through various architectures. The ANN model based forecasts were remarkably sharp as compared to other conventional models investigated in this study. As a result, the ANN modeling indicates a potential for great improvements in the state of rainfall forecasting.

Acknowledgments

On the outset I want to thank **Dr. Ashu Jain** for his support and guidance throughout my work. Inspite of his various involvements he was always available and patient. I am also greatly indebted to **Dr. T. Gangadharaiah** for his guidance and support. I'm also thankful to **Dr. Bithin Dutta** and **Dr. Rajesh Srivastava** for their kind cooperation and encouragement given to me, during my stay here at IIT Kanpur. I wish to thank my family for encouraging me to go for post-graduate studies. I would also like to thank my friends Atul Kumar, Kshitiz Krishna, Manoj Choudhary and Santosh Singh who had helped me with lots of ideas during the design phase.

My gratitude goes to all of my batch-mates, who made my stay here, in IITK, a memorable one. And last, but most certainly not the least, I acknowledge the M.Tech 1998 batch for the love and support they have lent me during our short meetings.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 General

The rainfall event is a part of atmospheric phenomenon, and the atmosphere is a non-linear and complex phenomenon. The formation of rainfall requires the lifting of an air mass in the atmosphere so that it cools and some of its moisture condenses. Surface heating of earth causes instability of moist air, and are sustained by the latent heat of vaporization given up as the water vapor rises and condenses. The condensation, often supported by dust floating in the air, give rise to the formation of tiny droplets which grow by impact with their neighboring droplets as they are carried by turbulent air motion, until they become large enough so that the force of gravity overcomes that of friction and they begin to fall, further increasing in size as they hit other droplets in the fall path.

Rainfall is the key input variable which activates flow and mass transport in hydrological systems, and models for simulating and forecasting rainfall in space and time can play an important role in enhancing our understanding of hydrological system response, and in the design and operation of water resources systems.

Forecasts of precipitation are important in a variety of contexts. An accurate

quantitative precipitation forecast (QPF) can identify the potential for heavy precipitation and possible associated flash flooding, as well as providing information for hydrologic interests. Placement and timing of QPF output can have significant impacts on river forecasts. As part of the modernization of the National Weather Service (NWS), more emphasis is being placed on the local generation of QPFs and their subsequent use in hydrological models at River Forecast Centers. The need for accurate local rainfall prediction is also apparent when considering other benefits such as reservoir operations and forestry interest informations.

A rainfall event indicates complex behavior as values of time-series, if the system has a small number of variables. As prediction lead-time increases, one would expect the system to be characterized by a decrease of the correlation between predicted and actual values. The conventional prediction method for the Quantitative Precipitation Forecasting have a limit of the lead-time decided by this property. It becomes too difficult to predict rainfall intensity for long lead-time by using meteorological equation with many kinds of observed data. Because atmospheric dynamic systems are sensitive to initial conditions. With respect to precipitation forecast systems, the declination of scope requires a decision. A narrow perspective might include only those studies in which forecasts were produced operationally, or at least were simulated under field like conditions, and were verified against actual observations. A broad perspective might include all science of hydrometerology that contributes the knowledge of forecasting.

A basic consistent mechanism needed for developing rainfall model is provided by *Reynolds transport theorem*, also known as the *general control volume theorem*. Reynolds transport theorem takes physical laws that are normally applied to a discrete mass of a substance and applies them instead to a fluid flowing continuously through a control volume. For this purpose, two types of fluid properties can be distinguished; *extensive* properties, whose value depend on the amount of mass present, and *intensive* properties, which are independent of mass present. For any *extensive* property B, a corresponding *intensive* property β can be defined as the quantity of

B per unit mass of fluid, that is $\beta = dB/dm$.

The Reynolds transport theorem relates the time rate of change of an extensive property in the fluid, dB/dt, to the external causes producing this change. The governing equation of this theorem is given as follows:

$$\frac{dB}{dt} = \frac{d}{dt}\int\int\int_{c.v.} \beta \rho dV + \int\int_{c.s.} \beta \rho \mathbf{V}.\mathbf{dA} \tag{1}$$

where ρ represents density of fluid

dV represents small element of control volume (c.v)

$\mathbf{V}$ represents the velocity vector and

$\mathbf{dA}$ represents the normal area vector within the c.v.

In words Reynolds transport theorem states that the *total rate of change of an extensive property of a fluid is equal to the rate of change of extensive property stored in the control volume,* $d/dt\int\int\int\beta\rho dV$, plus the net *outflow of extensive property through the control surface,* $\int\int\beta\rho\mathbf{V}.\mathbf{dA}$. Reynolds transport theorem may be applied to develop a model for rainfall forecasting as given below.

Taking a cylindrical column of atmosphere as a central volume and mass of water (*both in liquid and vapor phases*) as the extensive property, one can apply the *Reynolds transport theorem* and obtain an expression for the rainfall intensity falling on the ground from this *control volume.* Such a *control volume* is known as the "thunderstorm cell" and the resulting equations give rise to what is known as the "Thunderstorm Cell model" for rainfall forecasting. Thunderstorm Cell model, derived using Reynolds transport theorem, provides a suitable mechanism to model the movement of atmospheric moisture starting from lifting of an air mass in the thunderstorm cell to the formation and occurrence of precipitation. Thunderstorm Cell model is discussed in detail in the Chapter 3.

Statistical techniques such as regression and time-series analysis have been used to forecast the short-term rainfall using the available meteorological observations.

The water resources systems research laboratory (WRSRL) has made substantial progress in the development and application of stochastic point process and rainfall field models. Two rainfall modeling systems have evolved from this work: (i) a rainfall time series analysis and simulation package (RAINSIM), which is suitable for hydrological studies requiring long generated time series at one or more sites, and (ii) a stochastic space-time rainfall field modeling system; modified turning bands (MTB) which can be used for the simulation and forecasting of frontal rainstorms.

More recently Artificial Neural Networks (ANNs) have been proposed as a technique for modeling and forecasting. ANNs have been used in modeling a wide variety of physical systems including short-term rainfall forecasting. An ANN is based on past experience (real rainfall results in previous years), deviation of rainfall estimates with respect to real rainfall events, actual satellite data and partially known satellite data. While the data required to make such predictions have been available for quite some time, the complex, ever-changing relationships among the data and its effect on the probability, much less the quantity, of rain has often proved difficult using conventional computer analysis. The use of an ANN , however, which learns rather than analyzes these complex relationships, has shown a great deal of promise in accomplishing the goal of predicting both the probability and quantity of rain in a local area.

Simply stated, an ANN is a processing or computing device. Each network has many units, each with some local memory. The units are connected by communication channels, or connections. Each connection carries data to other units. These Networks need to be "trained" by examples and repetitions. After a Network has been trained and tested to the users satisfaction, it is ready for use. New sets of input data can be presented to the network, and it will present a forecast based on what it has learned. These Networks can be used to extract patterns and detect trends that are often too complex to be noticed by either humans or other computing devices.

1.2 Objectives of The Present Study

The primary objective of this study is to investigate the new technique of ANN for forecasting hourly rainfall. The project examines rainfall in the context of current available theoretical and mathematical models and will develop an ANN model to facilitate the forecasting of rainfall from sampled or averaged measurements, for use in climate and hydrological modeling. The process of developing an ANN model will require the following steps:

- Develop a computer program to simulate an ANN.
- Verify the computer program using hypothetical and/or real data.
- Select the physical variables affecting rainfall intensity.
- Obtain meteorological observation including rainfall intensities.
- Investigate various ANN architectures to model the rainfall intensities.
- Identify the optimal ANN model for rainfall forecasting.

The secondary objective of this study is to analyze the performance of the developed ANN models with present theoretical and statistical models . This may be done after an appropriate comparison is made among these models.

The hidden and most important objective of this study is to make a contribution towards the area of ANN application in forecasting of rainfall and thus to improve climate and hydrological modeling. It is expected that this study will be proved to be a guideline for future research and development in the area of ANN application to water resources systems.

1.3 Organization of the Thesis

The Chapter 1 of this thesis describes the problem of rainfall forecast modeling in general, various available modeling techniques, the objectives and organization of

the thesis. The rest of this thesis is organized as follows. In Chapter 2, we discuss the literature available in area of rainfall forecasting models. In Chapter 3 the new ANN techniques is discussed in detail. The development of various types of model structures are discussed in Chapter 4. Discussion of results is taken up in Chapter 5. Finally Chapter 6, presents the conclusions and discusses the ongoing work suggesting some future extensions. References and appendices are provided at the end.

Chapter 2

Literature Review

Many models have been developed so for to forecast rainfall events. These include deterministic models such as thunderstorm cell model and stochastic models such as regression and time-series models and relatively new ANN models. This chapter highlights the various models in brief. Rainfall data is widely collected by rain gages for nonoverlapping intervals such as seconds, minutes, hours, days etc. Most of the models developed use such meteorological data for the calibration and validation of models.

The methods available for rainfall forecasting depend on the scale of interest. Both time and space scales must be considered and depend on the particular application. Rainfall forecasting models in use by agencies such as the United States (US) National Weather Service (NWS) or the European Meteorological Center (EMC) are typically valid on large scales relative to those of interest in the field of hydrology, where a description of the variability of rainfall on smaller space and time scales is required. The localization of the large- scale numerical model results can be attempted by the use of regression models. The Model Output Statistics (MOS) method documented in Glahn and Lowry (1972) and the regression techniques used in the very short-term forecasting of severe local storms and thunderstorms (National Weather Service, Meteorological Service Division (NWS-MDS), 1981), are two examples of such models used operationally.

In the 1980s research concentrated mostly on estimation of rainfall rates from radar reflectivities and on prediction of rainfall rates through projection of trends detected in a time series of reflectivity fields. Some of these efforts have been reviewed by Chen and Kavvas (1992) who introduced a radically new technique. Harnessing concepts from pattern recognition theory, the technique decomposes the radar image of the rain field into constituent polygonal contours, whose evolution in time and space is tracked through subsequent radar images and then projected into the future via an adaptive exponential smoothing scheme. By decomposing the projected contours at a desired lead time, a prediction of the rain field is obtained. The technique was validated for lead times up to 30 minutes on historical radar data from one storm with encouraging results.

Recently, methods of disaggregating daily rainfall into shorter time periods have been proposed (Hershenhorn and Woolhiser, 1987). These models attempt to disaggregate daily rainfall into sequence of showers. However they require dozens of parameters to disaggregate daily rainfall into individual storm. Based on this paper a stochastic rainfall model is presented (Zhiquan, Shafiqul and Elathir, 1994). In this paper instead of trying to reproduce the specific rainfall events, an attempt was made to capture the statistics of hourly rainfall from the observed daily rainfall statistics. Once the parameters are estimated, the simulation may be done for the sequence of rainfall events at any desired timescale (e.g. from seconds to days). The model has been shown to capture temporal and spatial structure of rainfall.

The potential utility of radar reflectivity as an additional input to a physically based spatially lumped rainfall model was explored by Georgakakos and Krajewski (1991). Their model was intended for making predictions of mean areal rainfall over river basins of the order of 100–1000 km, with the lead time of one hour. The question was by how much can the forecast uncertainty be reduced as a result of augmenting the model input with radar data. A comparison of forecast error variances obtained with and without radar data indicated that a reduction of 5–15 %

in variance could be attained.

Seo and Smith (1992) formulated a two-component model for prediction of convective rainfall under the radar umbrella. Their principal assumption was that the vertically integrated liquid water, as a function of time and space, is equal to the sum of a time-varying mean and a residual that varies in time and space. A physically based model was proposed to predict the mean using radar data, surface measurements of temperature, dew point temperature and pressure, and radiosonde profiles of environmental temperature and water vapor density. A statistical autoregressive model was used to predict the residual. Validation was limited to seven historical storms that also provided data for parameter estimation. Rainfall fields estimated from radar reflectivities were assumed to be the ground truth. Predictions were made every 10–12 minutes, for one hour ahead and an area of 80,000 km^2. Based on the mean square error criterion, the model forecasts outperformed, though not substantially, the advection forecasts (obtained via a translation of the current rainfall field, estimated from radar data at the forecast time, by the mean velocity vector one hour into the future).

Another predictive model was developed by French and Krajewski (1994). The model was based on the conservation of mass and momentum laws in which states and boundary conditions are parameterized directly in terms of radar reflectivity (a predictor of liquid water content), satellite infrared brightness (a predictor of cloud top temperature), and surface air temperature, dew point temperature and pressure. For applications, state dynamics are linearized and states are updated based on sensor measurements via a Kalman filter. In a verification study, reported by French *et al.* (1994) and using historical data from three storms, predictions of the rainfall rate were computed every 10–15 minutes for the lead time of one hour and an area of 1,70,000 km^2. According to the mean error, mean square error, and the correlation between forecasted and measured (via the same radar) rainfall rates, the model forecasts outperformed the persistence forecasts (obtained under the assumption that the currently observed rainfall will continue for one hour) and performed

somewhat better than the advection forecasts.

Recently, new technique of ANNs has been applied to a wide variety of problems, including cloud classification, climatic warning and rainfall forecasting in a meteorological context. An ANN is a computational paradigm which implements simplified models of their biological counterparts, biological neural networks. They have great potential for solving complex problems such as systems control, data compression, optimization problems, pattern recognition, and system identification. Today, more research is directed towards the development of artificial neural networks for applications such as optimization, pattern matching, system modeling, function approximation, and control. Rainfall forecasting falls in the category of system modeling. A few rainfall forecasting models using ANN are found in literature. Following is the brief discussion about these models.

Multi-layer ANNs were introduced to solve the nonlinear rainfall forecasting problems (French, Krajewaski and Cuykendall, 1992). Rainfall fields were generated using a space-time mathematical rainfall simulation model, and forecasted fields were compared with the perfectly known model-produced fields. One hour ahead forecasts were produced and comparison with true mean areal intensities and percent areal coverage indicate that method performs well and may be trained for wide area and longer span of time.

Two types of ANNs were proposed by Moriyama and Muneo (1994) to predict rainfall time-series. One was Multi-layer neural networks (MLNN) and other was Elman recurrent neural networks (ERNN). The former one is conventional one while in ERNN, only the values of hidden layer of feedback are processed to same units at next step without multiplied by weights. The values are saved at a context layer. The ERNN is easy to implement extension from MLNN because both have same training phenomenon.

somewhat better than the advection forecasts.

Recently, new technique of ANNs has been applied to a wide variety of problems, including cloud classification, climatic warning and rainfall forecasting in a meteorological context. An ANN is a computational paradigm which implements simplified models of their biological counterparts, biological neural networks. They have great potential for solving complex problems such as systems control, data compression, optimization problems, pattern recognition, and system identification. Today, more research is directed towards the development of artificial neural networks for applications such as optimization, pattern matching, system modeling, function approximation, and control. Rainfall forecasting falls in the category of system modeling. A few rainfall forecasting models using ANN are found in literature. Following is the brief discussion about these models.

Multi-layer ANNs were introduced to solve the nonlinear rainfall forecasting problems (French, Krajewaski and Cuykendall, 1992). Rainfall fields were generated using a space-time mathematical rainfall simulation model, and forecasted fields were compared with the perfectly known model-produced fields. One hour ahead forecasts were produced and comparison with true mean areal intensities and percent areal coverage indicate that method performs well and may be trained for wide area and longer span of time.

Two types of ANNs were proposed by Moriyama and Muneo (1994) to predict rainfall time-series. One was Multi-layer neural networks (MLNN) and other was Elman recurrent neural networks (ERNN). The former one is conventional one while in ERNN, only the values of hidden layer of feedback are processed to same units at next step without multiplied by weights. The values are saved at a context layer. The ERNN is easy to implement extension from MLNN because both have same training phenomenon.

Tommy, Chow and Cho (1997) described a new approach consisting of extraction of the information from radar images and evaluating past rain gauge records to provide short term rainfall forecasting. All meteorology data were provided by the Royal Observatory of Hong Kong (ROHK). Pre-processing procedures were essential for this neural network rainfall nowcasting. The network architecture was based on a recurrent Sigma-Pi network. The results were very promising and this neural based rainfall forecasting system was capable of providing a reliable rain storm warning signal to the Hong Kong public in advance. Few more successful applications include Dimitris *et al.* (1997) and McCullagh (1997).

It is obvious that ANN is an emerging tool in rainfall forecasting using meteorological data. This study demonstrates the great potential of ANN in forecasting hourly rainfall from meteorological data.

Chapter 3

Artificial Neural Networks

3.1 Introduction

Artificial Neural Networks (ANN), commonly referred to as "neural networks", have been motivated right from their inception by the recognition that the brain computes in an entirely different way from the conventional digital computers. The struggle to understand the brain owes much to the pioneering work of Ramon Y. Cajal (Ramon,1911), who introduced the idea of neurons as structural constituents of the human brain. A neuron is the smallest processing unit in a human brain which is about five to six times slower than the smallest processing unit in a computer. However, the brain makes up for the relatively slow rate of operation of a neuron by having a truly staggering number of neurons (nerve cells) with massive interconnection between them; it is estimated that there must be on the order of 10 billion neurons in the human cortex, and 60 trillion synapses or connection (Shepherd, 1990). The net result is that the brain is an enormously efficient structure. Specifically, the *energetic efficiency* of the brain is approximately 10^{-16} joules(J) per operation per second, whereas the corresponding value for the best computers in use today is about 10^{-6} joules per operation per second (Faggin,1991). Figure(1) shows the structural organization levels of brain (Haykin,1994).

In its most general form, an ANN is a machine that is designated to model the

way in which the brain performs a particular task or function of interest; the network is usually implemented using electronic components or simulated in software on a digital computer. To achieve good performance, ANNs employ a massive interconnection of simple computing cells referred as "neurons" or "processing units". Following is the definition of an ANN viewed as an adaptive machine (Alek,1990):

Artificial Neural Network is a massive parallel distributed processor that has a natural propensity for strong experiential knowledge and making it available for use.

It resembles the brain in two respects:

- Knowledge is acquired by the network through a learning process.
- Inter-neuron connection strengths known as synaptic weights are used to store the knowledge.

The procedure used to perform the learning process is called a *learning algorithm*, the function of which is to modify the synaptic weights of the network in an orderly fashion so as to attain a desired design objective. ANNs are also referred to in the literature as *neurocomputers, connectionist networks, parallel distributed processors etc.* ANNs have following properties affiliated with them:

3.1.1 Nonlinearity

The nonlinearity of an ANN is of interest in relation with the data (e.g. rainfall intensity) it has to classify. An ANN is a set of interconnections of artificial neurons; which possess nonlinearity in its basic structure and therefore the nonlinearity thus projected by it is spread throughout the whole network.

3.1.2 Input-Output Mapping

It is a process of forcing an ANN to yield a particular response to a specific input. A particular response may or may not be specified to provide external correction. The algorithm of back propagation is used for learning multi-layer ANNs with sequential

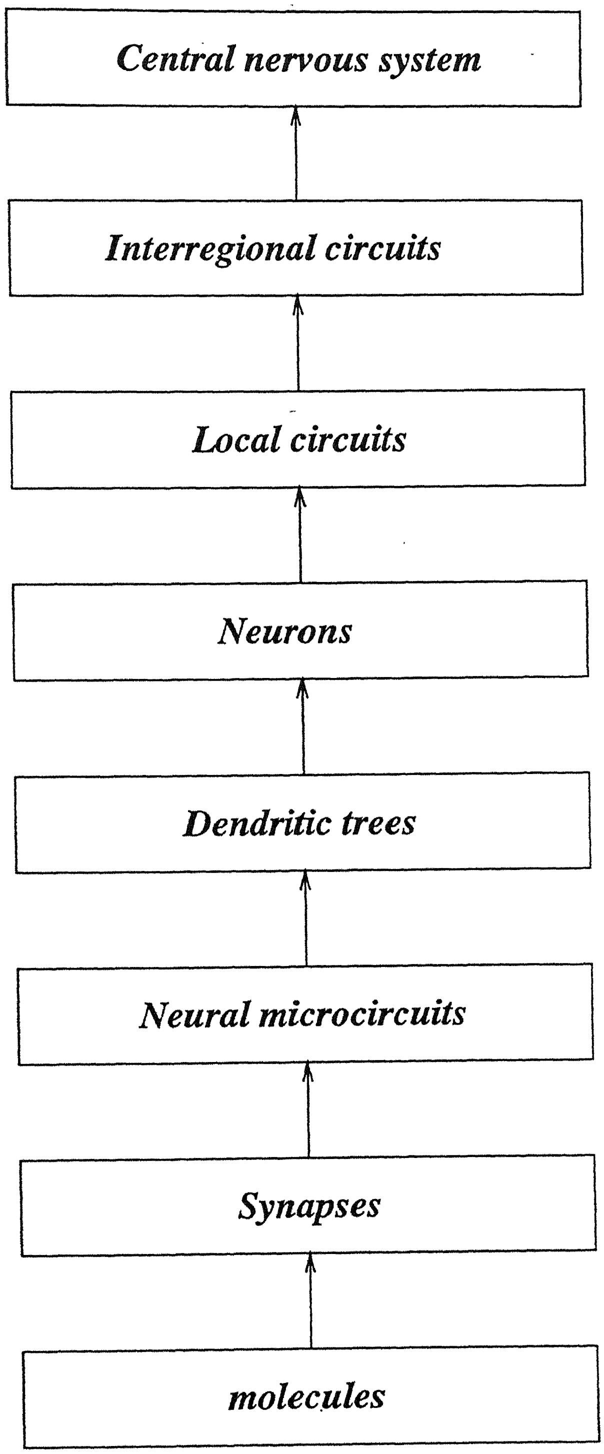


Figure 1: Structural organization of levels in the brain

links. On each neuron of the first layer all units of an external signal move. The neurons execute weighed summation of units of input signals. To a sum of units of input signals, multiplied on an appropriate synaptic weight, offset of a neuron increases. Above (over) an outcome of summation, a nonlinear conversion - function of activation (transfer function) is executed. The value of the function of activation is output of a neuron. Learning with supervision modifies the weights by a training set to minimize the difference between the desired and actual response until an appropriate statistical criterion is satisfied. Thus an ANN captures the learning steps through a cause-effect relationship and this input-output mapping is needed to perform the processing tasks.

3.1.3 Adaptivity

One of the distinct strengths of ANNs is their ability to generalize. The network is said to be generalized well when it sensibly interpolates input patterns that are new to the network. An ANN has an inherent capability to adapt their weights to changes in surrounding environment. Adaptivity makes it possible to change the weights of a suggested network within a non-stationary environment. The generalized ANN models thus obtained are very much useful in flood and rainfall related complex applications.

3.1.4 Neurobiological Analogy

The design of an ANN is motivated by analogy with the brain, which is living proof that fault-tolerant parallel processing is not only physically possible but also fast and powerful. ANNs are thus inspired by knowledge from neuroscience, that consist of massively interconnected processing units (also called as neurons). Their massively interconnected architecture has the potential for processing different types of dataset, they may be probabilistic, redundant, noisy, inconsistent etc. They can, in principle, easily adapt to a new environment by learning, which also forms the basis to solving problems, as here done in Rainfall forecasting.

3.2 ANN Architecture

The pattern in which the neurons of an ANN are structured is designated by the term ANN architecture. In this section, various ANN architectures (structures) have been discussed.

3.2.1 Single-Layer ANNs

In a *layered* ANN the neurons are connected in a layered structure. There is full connectivity between the layers but no connections within a layer or between two layers that are not adjacent. The simplest form of a layered network consists of an *input layer* of source nodes that projects onto an *output layer* of neurons, but not vice versa. This network is basically a *feed-forward* type network. Figure (2) depicts a *single layer feed-forward network* with an input layer X_i and an output layer Y_i. The weights between the input layer and the hidden layer are w_{ij}. However if the computations are performed at the *output layer* also then a *two-layer feed-forward network* is performed with an additional hidden layer.

3.2.2 Multi-Layer ANNs

The second class of a feed-forward network which is used in the present work distinguishes itself by the presence of one or more *hidden layers*, whose computation nodes are correspondingly called *hidden neurons* or *hidden units*. The function of the hidden neurons is to intervene between the external input and the network output. By adding one or more hidden layers, the network is enabled to extract higher-order statistics, because the network acquires a *global* perspective despite its local connectivity by virtue of the extra set of synaptic connections and the extra dimension of neural interactions (Church, 1992). The ability of hidden neurons to extract higher-order statistics is particularly valuable when the size of the input layer is large.

The source nodes in the input layer of the network supply respective elements of the activation pattern (input vector), which constitute the input signals applied

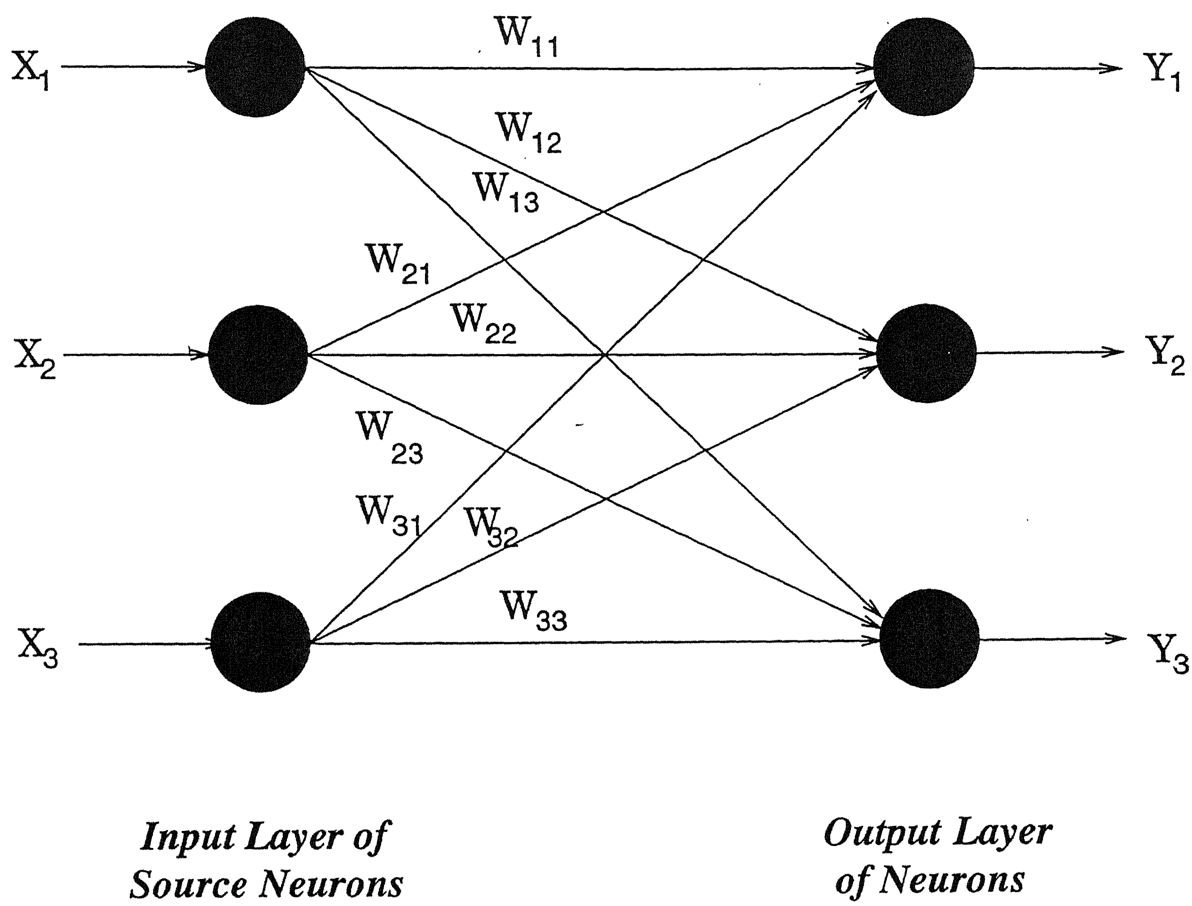


Figure 2: Feed-forward Network with a Single-layer of Neurons

to the neurons (computation nodes) in the second layer (i.e. the first hidden layer). the output signals of the first hidden layer are used as input to the third layer, and so on for the rest of network. Typically, the neurons in each layer of the network have as their inputs the output signals of the preceding layer only. The set of output signals of the neurons in the output (final) layer of the network constitutes the overall response of the network to the activation pattern supplied by the source nodes in the input (first) layer. The network shown in Figure (3) is said to be *fully connected* in the sense that every node in each layer of the network is connected to every other node in the adjacent forward layer.

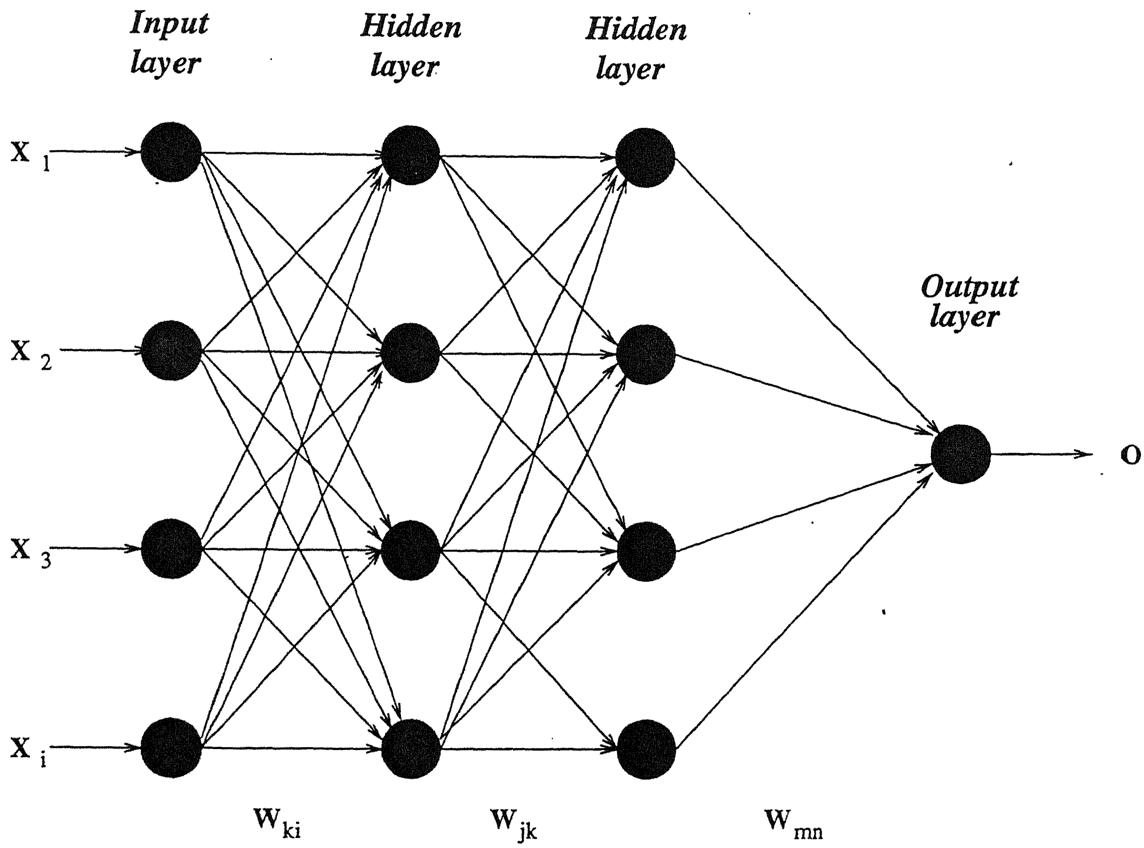


Figure 3: Multi-layer Feed-forward ANN

Multilayer feed-forward ANNs can be successfully applied to modeling of a variety of physical systems. It is not necessary to know a formal mathematical model of the problem to train and then recall information from the feed-forward neural system. Instead, if a sufficient training set and a suitable network architecture are devised, then the error back-propagation algorithm can be used to adapt network parameters to obtain an acceptable solution. *The solution is obtained through experimentation and simulation rather than through rigorous and formal approach to the problem.* As such, ANN computation offers techniques that are in the middle ground, somewhere between the traditional engineering and the artificial intelligence

approach. Rumelhart (Rumelhart *et. al.* 1986) described a learning procedure for layered ANN through back-propagation algorithm which is described in following section.

3.2.3 Back-propagation algorithm

Rumelheart (Rumelhart, 1986) first integrated the ideas of *perceptron* and *least mean square error* in the form of a learning algorithm, which is known as *back propagation* model of an ANN. The weights of interconnections of the neurons for the perceptron model are decided by minimizing the error between the desired and actual output of ANN and the process is known as *training* or *learning* of the ANN. The minimization of error, in other words the modification of weights is done using steepest descent method of optimization. The input I_i to any neuron i is given by

$$I_i = \sum_{i=1}^{N} O_j W_{ji} \tag{2}$$

where O_j is the input activation from unit i and W_{ji} weight connecting unit i to unit j and N is no. of neurons in the previous layer. However, instead of calculating a binary output, the net input is added to the unit's bias and the resulting value is passed through a sigmoid function:

$$O_i = \frac{1}{1 + exp(-I_i x)} \tag{3}$$

where x is an integer positive number.

According to Equation (2), the dependence of O_i on I_i is sigmoidal with the linear region of slope $(1/x)$. The weighted sum of all these outputs acts as the input to the neurons of subsequent layers. In the back-propagation method, the least square error is defined by:

$$E = \sum_{i=1}^{N} (O_i - D_i)^2 \tag{4}$$

where O_i = Observed output from the ith neuron in the output layer of the ANN
D_i = Desired output at the ith neuron in the output layer of the ANN

N = Total no. of neurons in the previous layer

Learning in a back-propagation network is in two steps. First, each pattern is presented to the network and propagated forward to the output. Second, a method called gradient descent is used to minimize the total error on the patterns in the training set. Initially all the weights are randomized to some initial value. After the error on each pattern is computed, each weight is adjusted in proportion to the calculated error gradient and is back-propagated from the outputs to the inputs. The changes in the weights reduce the overall error in the network. Thus, the appropriate change in the weight can be written as

$$W_{ji}(n+1) = W_{ji}(n) + \Delta W_{ji}(n) \tag{5}$$

$$\Delta W_{ji}(n) = \beta(\delta_i)O_j - \alpha \Delta W_{ji}(n-1) \tag{6}$$

where $\Delta W_{ji}(n+1)$ is the updated value of weight W_{ji} at $(n+1)^{th}$ iteration

$\Delta W_{ji}(n)$ is the change in weight W_{ji} at n^{th} iteration

$\Delta W_{ji}(n-1)$ is the change in weight W_{ji} at $(n-1)^{th}$ iteration

β is learning constant

α is momentum constant, and

δ_i is an error signal defined as follows

For the neuron in the output layer:

$$\delta_i = O_i(1 - O_i)(D_i - O_i) \tag{7}$$

In multilayer neural networks target output for hidden layers is unknown. So the weighted error signals of all the neurons of the succeeding layers is utilized to calculate error signal of any neuron i in the hidden layer and is expressed as :

$$\delta_i = O_i(1 - O_i) \sum_{m=1}^{M} \delta_m W_{mi} \tag{8}$$

where m runs over all the neurons (*i.e.* M) in the subsequent layer. This value of δ_i is utilized to calculate the change in weights and the procedure of updating the coefficients W_{ji} is repeated until the errors are converged up to an acceptable level.

3.3 Activation Function

The activation function denotes the output from a neuron in terms of the activity level as its input. The choice of activation function can change the behavior of the ANN network considerably. The standard choice is the sigmoid function, though other activation functions are also in use viz., threshold functions, piecewise functions, hyperbolic functions etc. The sigmoid function are used either in symmetric[-1,1] or asymmetric [0,1] form. The sigmoid function is represented by following equation:

$$f(x) = \frac{1}{(1 + exp(-ax))} \tag{9}$$

where a is the *slope parameter* of the sigmoid function. In fact, the slope at the origin equals $a/4$. In the limit, as the slope parameter approaches infinity, the sigmoid function becomes simply a threshold function. Whereas a threshold function assumes the value of 0 or 1, a sigmoid function assumes a continuous range of values from 0 to 1.

3.4 Initial Weights

Now that we have waded through all of the details of the back-propagation learning equations, let us consider how we should choose the initial weights for our network. Suppose that all of the weights start out at equal values. If the solution to the problem requires that the network learn unequal weights, then having equal weights to start with will prevent the network from learning. This is the case, because of the fact that the error back-propagated through the network is proportional to the value of the weights. If all the weights are the same, then the back-propagated errors will be the same, and consequently all of the weights will be updated by the same amount. To avoid this symmetry problem, the initial weights to the network should be unequal.

Another issue to consider is the magnitude of the initial weights. When the randomized weights are selected, all of the network weights and biases are initialized

to small random values. As the weight updates in the back-propagation algorithm are proportional to the derivative of the sigmoid activation function, it is important to consider how the net input affects its value. The derivative is a maximum when $f(x)$ is equal to 0.5 and approaches its minimum as $f(x)$ approaches 0 or 1. Thus, weights will be changed most for unit activations closest to 0.5 (the steepest portion of the sigmoid). Once a unit's activation becomes close to 0 or 1, it is committed to being on or off, and its associated weights will change very little. It is therefore important to select small initial weights so that all of the units are uncommitted (having activations that are all close to 0.5; the point of maximal weight change). Normally it is suggested to initialize the weights within therange of $\pm0.3, \pm0.5$ or ±0.7 depending upon the applications.

3.5 Learning Constant

The effectiveness and convergence of the error back-propagation learning algorithm depends on the value of the learning constant β. In general, however, the optimum value of β depends on the problem being solved, and there is no single learning constant value suitable for different training cases. This problem seems to be common for all gradient-based optimization schemes. While gradient descent can be an efficient method for obtaining the weight values that minimize an error, error surfaces frequently posses properties that make the procedure slow to converge.

When broad minima yield small gradient values, then a larger value of β will result in a more rapid convergence. However, for problems with steep and narrow minima, a small value of β must be taken to avoid overshooting the solution. This concludes that β should be chosen experimentally for each problem. Past performances show that only small learning constants guarantee a true gradient descent. The price of this guarantee is an increased total number of learning iterations that need to be made to reach the satisfactory solution. It is also desirable to monitor the progress of learning by looking at the *error* versus *iteration* so that β can be increased at appropriate stages of training to speed up the minimum seeking.

Although the choice of the learning constant depends on the class of the learning problem and on the network architecture, the values ranging from $10^{-}13$ to 10 have been reported throughout the technical literature as successful for many computational back-propagation experiments. For large learning constants, the learning speed can be drastically increased; however, the learning may not be exact, with tendencies to overshoot, or it may never stabilize at any minimum, the same has been observed in the present work also.

3.6 Momentum Method

A momentum term is usually included in the equations used to train an ANN using back-propagation training algorithm. Although it is well known that such a term greatly improves the speed of learning, there have been few rigorous studies of its mechanisms. The momentum term improves the speed of convergence by adjusting the current weight with a fraction of the most recent adjustment.The optimal condition for convergence may be depicted by the following equation:

$$\Delta W_{ji}(n+1) = \beta(\delta_i)O_j - \alpha\delta_i W_{ji} \tag{10}$$

where $n+1$ and n indicates the current and the most recent training step, and α is a user-selected positive momentum constant. The second term is said to be a scaled most recent adjustment of weights and denoted as the *momentum term*. Typically, α is chosen between 0.1 and 0.8. The momentum term can increase the range of learning rate over which the system converges.

3.7 Applications of ANN in Engineering

Artificial Neural Networks have been successfully used in a wide variety of engineering applications. Some of these engineering applications include ecological modeling (Scardi,1998); daily electric load forecasting (Park *et al.*, 1991); time-series forecasting (Sharda and Patil, 1990); rainfall-runoff modeling (Hoshin, Gupta and

Soroosh,1993) and moreover in forecasting short-term (hourly) rainfall as described in Chapter 2.

Chapter 4

Model Development

Three different types of modeling techniques have been investigated for rainfall forecasting in this study. These include ANN, Regression Analysis and Thunderstorm Cell model. In addition, a Persistence approach is also used for the investigations.

The meteorological data at Indianapolis Eaglecreek airport for the year 1997 were downloaded from the internet (*http://shadow.agry.purdue.edu*). The data consists of rainfall, temperature, dew-point temperature and relative humidity for the whole year. All the values were observed at 1 hour interval. The concerned data are presented in Appendix A.

4.1 The ANN Model

The input variables considered here in forecasting Rainfall are Temperature, Relative Humidity and Dew-point Temperature, whereas the output variable is considered to be the Rainfall intensity. The development of an ANN model may be accomplished as per following steps:

Pre-Processing of Data

To increase the effectiveness of ANN learning it is necessary to perform some pre-processing of the data before presentation to the network. This is done by identifying most prominent variables on which the output neorodes (rainfall intensity) is dependent and normalizing the data available between the values 0.1 and 0.9.

Architecture Selection

After processing the dataset a suitable ANN architecture is chosen based on trial and error basis. An ANN architecture consists of an output layer containing equal number of neurons as the dependent variables, an input layer containing equal number of neurons as the variables on which the output (rainfall intensity) is dependent, and one or more hidden layers. An example set is then *trained* by using an appropriate algorithm; the process being called as *learning* of network.

Validation of ANN

Testing of the network is done by presenting the trained ANN with a different dataset which has never been introduced to the trained ANN. Evaluation is done by comparing the observed output and the output calculated so for by the proposed ANN model.

4.1.1 Development of Computer Code

A computer code in 'C' has been developed using back-propagation algorithm as the training mechanism to predict the rainfall intensity in this study. Figure 4 depicts the flow chart for back-propagation simulation in an ANN. The learning ends when all the patterns in the training set have been presented to the ANN.

4.1.2 Verification of the computer Code

The computer code is verified through various real and available datasets. In preliminary stage data generated randomly for a straight line equation like $y = mx$

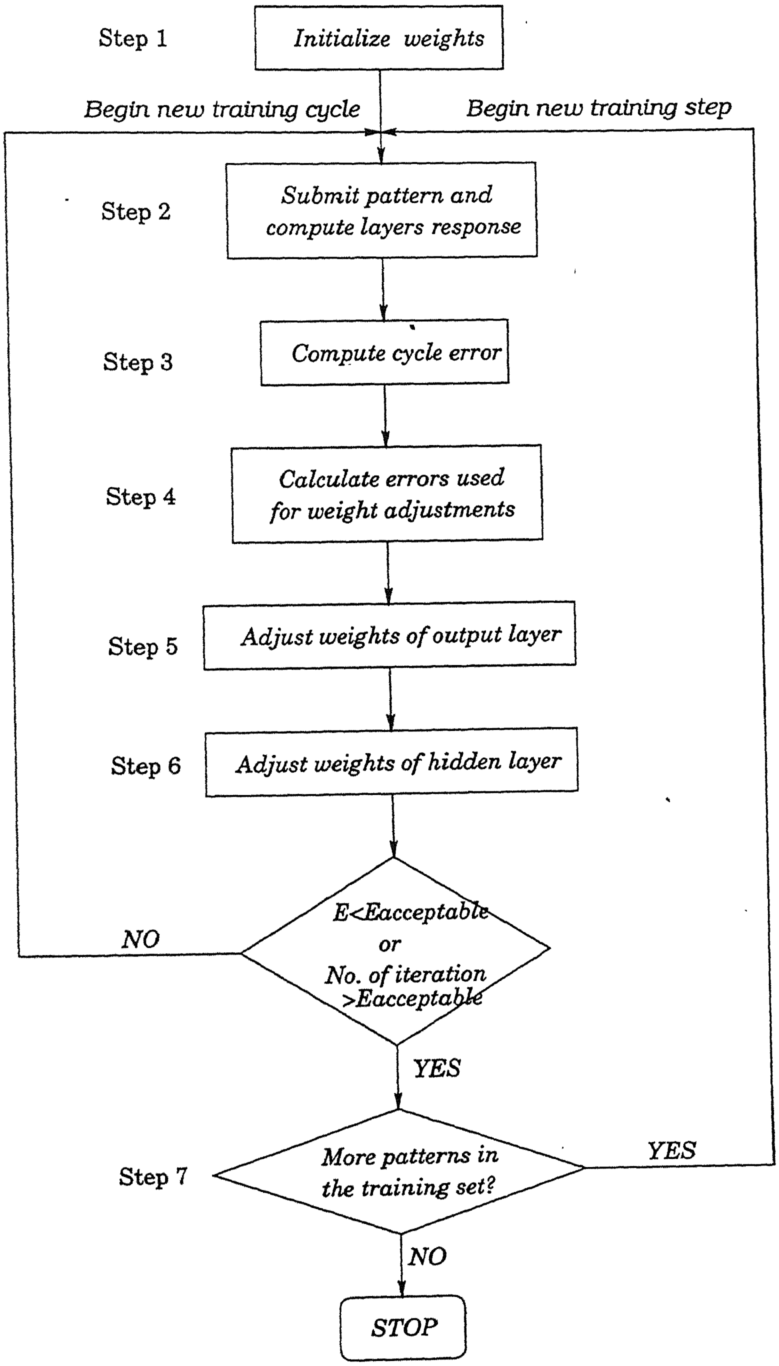


Figure 4: Flow Chart for the Computer Code

and the dataset so found were then perturbed slightly. These generated data were used to train a simple 1-2-1 ANN. This simple 1-2-1 architecture was giving a nice approximation with an average error of 2.07%. Again another curve $y = ax^2 + bx + c$ was utilized and the dataset so generated is given a perturbance. This dataset also verified for ANN code and the results were justified with an average error of 2.27%. Finally, a more realistic approach was made by considering a real dataset of a Steel Plant. The data used for the verification of computer program is daily operational data for Blast furnace at Visakhapatnam Steel Plant, Visakhapatnam, India (Punyasheel, M.Tech Thesis, (1999),"*Prediction of Coke Quality and Coke Rate using Fuzzy Logic and Genetic Algorithms* utilized for coke rate prediction modeling. The operational parameters are Hot blast temperature in *celcius* and Blast pressure in *atm.* and the dependent variable is Coke rate in *kg/m.tonne hour* The set of data and the statistical parameters for predicting coke rate are presented in Appendix B. The average error for 124 data was 4.76%.

4.1.3 ANN Model for Rainfall Prediction

A general code was written to train a network with any number of neurons in any layer of the structure for any number of layers using back-propagation training algorithm. In this study various ANN architectures have been investigated to predict the rainfall intensity for the Indianapolis Eaglecreek airport station, Indiana. The architecture with minimum average relative error is considered for further modification of more layers keeping in view that networks with smaller number of neurons in the hidden layer giving almost similar performance are given priority over the networks with large number of neurons in the hidden layer.

Initially simple three-layer ANN models with single hidden layer were developed for modeling the rainfall intensities which were later modified to a more complex four-layer ANN model with two hidden layers.

Three layer ANN Model

To proceed with a simple three-layer model a 7-N-1 ANN model was first investigated with 7 neurons in input layer, 3 to 25 neurons in hidden layer and 1 neuron at output layer. The 7 neurons in input layer represented temperature, dew-point temperature and relative humidity at time-step '*t*' and '(t-1)' and the rainfall intensity at time-step '(t-1)' respectively, while the output layer neuron represented the rainfall intensity at time-step '*t*'. Later, 11-N-1 and 15-N-1 structured models were investigated considering '(t-2)' and '(t-1)' time steps back neurons also in input layer. The exhaustive results in terms of statistical parameters from these models are presented in Appendix C. The numeric and graphical representation of best results is shown in Chapter 5. However, the best models were found to be 7-13-1, 11-18-1 and 15-18-1. Above all model 15-18-1 is proved to be the most suitable for predicting rainfall. Figure 5 depicts the three-layer ANN model for rainfall prediction. The detailed results of various ANN architectures is given in Appendix 'C'.

Thus reasonable performance was shown by three-layer models itself depending on the number of neurodes in input layer. Even it is found that a four-layer ANN model can propose much better relationship.

Four layer ANN Model

To attain better results one more hidden layer was inserted between input and output layers. Thus this architecture had one input layer, one output layer and two hidden layers containing same operational parameters as discussed above. Figure 6 shows the four-layer ANN model for rainfall prediction. The exhaustive results in terms of statistical parameters from these models are presented in Appendix 'C'. The numeric and graphical representation of best results is also shown in Chapter 5. However the best models were found to be 7-10-5-1, 11-13-5-1 and 15-11-8-1. Above all model 15-11-8-1 is proved to be the most suitable for predicting rainfall. The detailed results of various ANN architectures is given in Appendix 'C'.

To observe the performance of the ANN, initially same dataset was used for

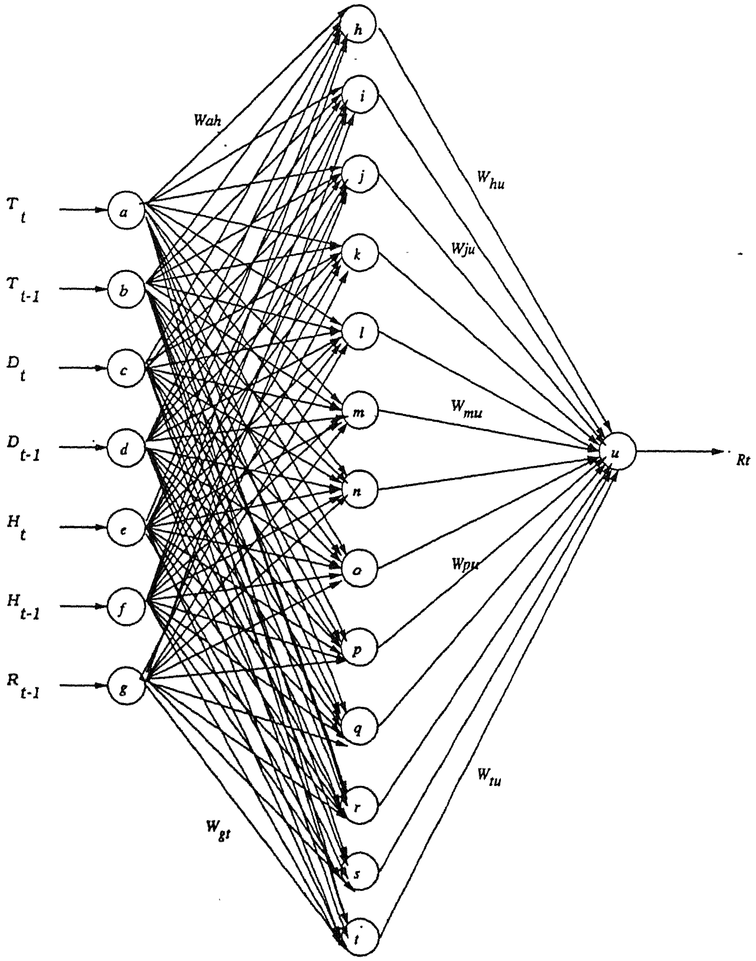


Figure 5: Three-layer ANN model for Rainfall Prediction

training as well as in testing. How well the trained ANN was recognizing the patterns which it had already seen before was judged from the results. The architecture thus obtained were considered as basic model for further investigation. In this study, it was observed that the ANN has memorized the 212 sets remarkably well, which means the modification of weights proceeded in the right direction. Then data were divided in two sets, one is for training (166 data points) and other is for testing (46 data points). The Optimum ANN structure for the present problem may be summarized as in Table 1. It was observed that in the initial stages of learning the error reduction was remarkable and ANN stopped the learning as the error

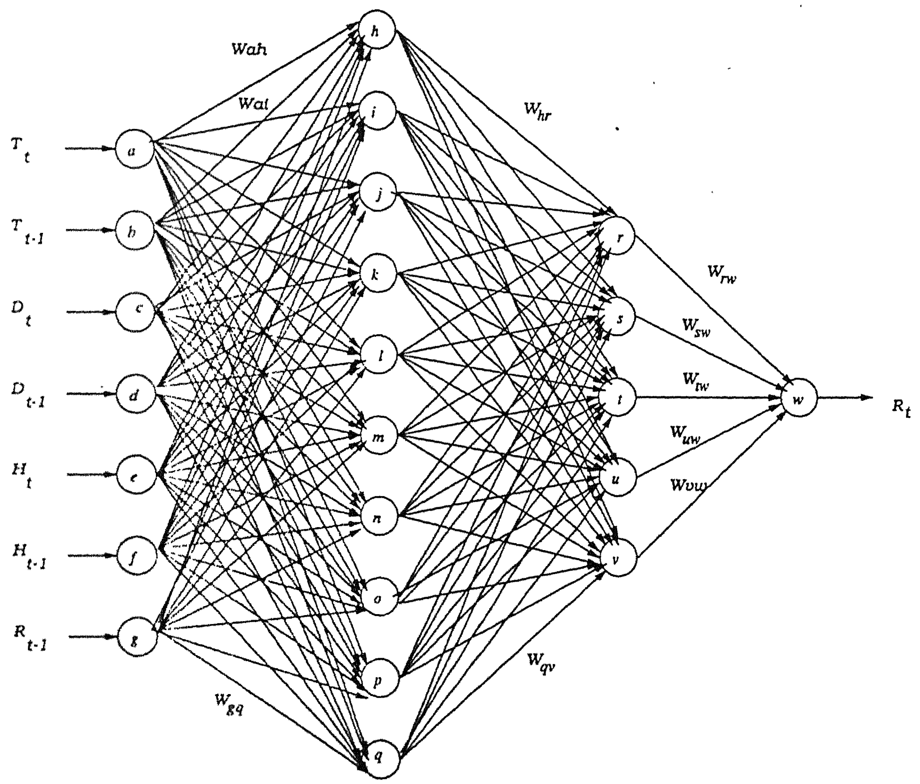


Figure 6: Four-layer ANN model for Rainfall Prediction

surface flattened. In this study the performance was first checked using same dataset for training and testing and with the justified observation of pattern recognition capabilities of ANN the dataset is further divided in two sets, one for training (167 data points) and other for testing (45 data points). The results as said above are presented in the Appendix 'C' and the Optimum models are discussed in next chapter.

4.2 Regression Models

Regression is a mathematical measure of the average relationship between two or more variables in terms of the original units of the data. In this section various regression models are investigated to predict rainfall intensity considering both *linear* and *non-linear* regressions.

Parameters	Values
Number of neurons in input layer	15
Number of neurons in 1st hidden layer	11
Number of neurons in 2nd hidden layer	8
Number of neurons in output layer	1
Learning rate Constant β	0.8
Momentum Constant α	0.2

Table 1: Optimum ANN structure of the Four layer ANN model

4.2.1 Linear Multiple Regression Models

Three linear multiple regression models were investigated to model the rainfall intensity using aforesaid 7, 11 and 15 operational variables. The structures of these three models are as follows:

Linear Multiple Regression Model(LMRM-1)

This model consisted of 7 operational variables which are self-explanatory. The structure of this linear multiple regression model is described as follows:

$$R_t = \alpha_0 + \alpha_1 T_t + \alpha_2 T_{t-1} + \alpha_3 D_t + \alpha_4 D_{t-1} + \alpha_5 H_t + \alpha_6 H_{t-1} + \alpha_7 R_{t-1} \tag{11}$$

where R's represent rainfall intensity

α's represent regression coefficients to be determined

T's represent temperature

D's represent dewpoint temperature

H's represent relative humidity and

t represents time- step

Linear Multiple Regression Model(LMRM-2)

This model consisted of 11 operational variables which are self-explanatory. The structure of this linear multiple regression model is described as follows:

$$
\begin{aligned}
R_t &= \alpha_0 + \alpha_1 T_t + \alpha_2 T_{t-1} + \alpha_3 T_{t-2} + \alpha_4 D_t + \alpha_5 D_{t-1} + \alpha_6 D_{t-2} \\
&\quad + \alpha_7 H_t + \alpha_8 H_{t-1} + \alpha_9 H_{t-2} + \alpha_{10} R_{t-1} + \alpha_{11} R_{t-2} \qquad (12)
\end{aligned}
$$

Linear Multiple Regression Model(LMRM-3)

This model consisted of 15 operational variables which are self-explanatory. The structure of this linear multiple regression model is described as follows:

$$
\begin{aligned}
R_t &= \alpha_0 + \alpha_1 T_t + \alpha_2 T_{t-1} + \alpha_3 T_{t-2} + \alpha_4 T_{t-3} + \alpha_5 D_t + \alpha_6 D_{t-1} + \alpha_7 D_{t-2} \\
&\quad + \alpha_8 D_{t-3} + \alpha_9 H_t + \alpha_{10} H_{t-1} + \alpha_{11} H_{t-3} + \alpha_{12} H_{t-3} + \alpha_{13} R_{t-1} + \alpha_{14} R_{t-2} \\
&\quad + \alpha_{15} R_{t-3} \qquad (13)
\end{aligned}
$$

The regression coefficients of the linear regression models above were computed using standard regression and gauss elimination programs. The regression coefficients for various models are summarized in Table 2. Other statistical parameters like Correlation Coefficient and Threshold statistics are discussed in Chapter 5.

4.2.2 Non Linear Multiple Regression Models

Six nonlinear multiple regression models were investigated to model the rainfall intensities using aforesaid 7, 11 and 15 operational parameters. The first three models were nonlinear models of order two whereas the next three were nonlinear models of order three. The structures of these nonlinear regression models are as follows:

Non Linear Multiple Regression Model (NLMRM-1)

$$
\begin{aligned}
R_t &= \alpha_0 + \alpha_1 T_t + \alpha_2 T_{t-1} + \alpha_3 D_t + \alpha_4 D_{t-1} + \alpha_5 H_t + \alpha_6 H_{t-1} + \alpha_7 R_{t-1} \\
&\quad + \beta_1 T_t^2 + \beta_2 T_{t-1}^2 + \beta_3 D_t^2 + \beta_4 D_{t-1}^2 + \beta_5 H_t^2 + \beta_6 H_{t-1}^2 + \beta_7 R_{t-1}^2 \qquad (14)
\end{aligned}
$$

Parameters	LMRM-1	LMRM-2	LMRM-3
α_0	0.0051	-0.0024	0.0143
α_1	-0.0032	0.0142	0.0021
α_2	-0.0075	0.0035	-0.0123
α_3	0.0024	-0.0047	0.0156
α_4	0.0057	0.0053	-0.0038
α_5	0.0049	-0.0038	0.0046
α_6	-0.0137	0.0141	0.0058
α_7	0.1642	-0.0047	-0.0029
α_8		0.0063	0.0053
α_9		-0.0152	0.0036
α_{10}		-0.0587	-0.0072
α_{11}		0.1645	0.0043
α_{12}			-0.0157
α_{13}			0.1723
α_{14}			-0.0957
α_{15}			0.1848

Table 2: Regression Coefficients of LMRM-1, LMRM-2 and LMRM-3 Models

Non Linear Multiple Regression Model (NLMRM-2)

$$\begin{aligned} R_t &= \alpha_0 + \alpha_1 T_t + \alpha_2 T_{t-1} + \alpha_3 T_{t-2} + \alpha_4 D_t + \alpha_5 D_{t-1} + \alpha_6 D_{t-2} + \alpha_7 H_t \\ &\quad + \alpha_8 H_{t-1} + \alpha_9 H_{t-2} + \alpha_{10} R_{t-1} + \alpha_{11} R_{t-2} + \beta_1 T_t^2 + \beta_2 T_{t-1}^2 + \beta_3 T_{t-2}^2 \\ &\quad + \beta_4 D_t^2 + \beta_5 D_{t-1}^2 + \beta_6 D_{t-2}^2 + \beta_7 H_t^2 + \beta_8 H_{t-1}^2 + \beta_9 H_{t-2}^2 + \beta_{10} R_{t-1}^2 \\ &\quad + \beta_{11} R_{t-2}^2 \end{aligned} \tag{15}$$

Non Linear Multiple Regression Models(NLMRM-3)

$$\begin{aligned} R_t &= \alpha_0 + \alpha_1 T_t + \alpha_2 T_{t-1} + \alpha_3 T_{t-2} + \alpha_4 T_{t-3} + \alpha_5 D_t + \alpha_6 D_{t-1} + \alpha_7 D_{t-2} \\ &\quad + \alpha_8 D_{t-3} + \alpha_9 H_t + \alpha_{10} H_{t-1} + \alpha_{11} H_{t-2} + \alpha_{12} H_{t-3} + \alpha_{13} R_{t-1} + \alpha_{14} R_{t-2} \\ &\quad + \alpha_{15} R_{t-3} + \beta_1 T_t^2 + \beta_2 T_{t-1}^2 + \beta_3 T_{t-2}^2 + \beta_4 T_{t-3}^2 + \beta_5 D_t^2 + \beta_6 D_{t-1}^2 \\ &\quad + \beta_7 D_{t-2}^2 + \beta_8 D_{t-3}^2 + \beta_9 H_t^2 + \beta_{10} H_{t-1}^2 + \beta_{11} H_{t-2}^2 + \beta_{12} H_{t-3}^2 + \beta_{13} R_{t-1}^2 \\ &\quad + \beta_{14} R_{t-2}^2 + \beta_{15} R_{t-3}^2 \end{aligned} \tag{16}$$

The above three equations model the rainfall intensities using $7, 11$ and 15 operational parameters. The results are discussed and compared in the next chapter. Coefficients obtained in these nonlinear models are summarized in table 3.

Non Linear Multiple Regression Models(NLMRM-4)

$$\begin{aligned} R_t &= \alpha_0 + \alpha_1 T_t + \alpha_2 T_{t-1} + \alpha_3 D_t + \alpha_4 D_{t-1} + \alpha_5 H_t + \alpha_6 H_{t-1} + \alpha_7 R_{t-1} \\ &\quad + \beta_1 T_t^2 + \beta_2 T_{t-1}^2 + \beta_3 D_t^2 + \beta_4 D_{t-1}^2 + \beta_5 H_t^2 + \beta_6 H_{t-1}^2 + \beta_7 R_{t-1}^2 \\ &\quad + \gamma_1 T_t^3 + \gamma_2 T_{t-1}^3 + \gamma_3 D_t^3 + \gamma_4 D_{t-1}^3 + \gamma_5 H_t^3 + \gamma_6 H_{t-1}^3 + \gamma_7 R_{t-1}^3 \end{aligned} \tag{17}$$

Non Linear Multiple Regression Models(NLMRM-5)

$$\begin{aligned} R_t &= \alpha_0 + \alpha_1 T_t + \alpha_2 T_{t-1} + \alpha_3 T_{t-2} + \alpha_4 D_t + \alpha_5 D_{t-1} + \alpha_6 D_{t-2} + \alpha_7 H_t \\ &\quad + \alpha_8 H_{t-1} + \alpha_9 H_{t-2} + \alpha_{10} R_{t-1} + \alpha_{11} R_{t-2} + \beta_1 T_t^2 + \beta_2 T_{t-1}^2 + \beta_3 T_{t-2}^2 \\ &\quad + \beta_4 D_t^2 + \beta_5 D_{t-1}^2 + \beta_6 D_{t-2}^2 + \beta_7 H_t^2 + \beta_8 H_{t-1}^2 + \beta_9 H_{t-2}^2 + \beta_{10} R_{t-1}^2 \\ &\quad + \beta_{11} R_{t-2}^2 + \gamma_1 T_t^3 + \gamma_2 T_{t-1}^3 + \gamma_3 T_{t-2}^3 + \gamma_4 D_t^3 + \gamma_5 D_{t-1}^3 + \gamma_6 D_{t-2}^3 \\ &\quad + \gamma_7 H_t^3 + \gamma_8 H_{t-1}^3 + \gamma_9 H_{t-2}^3 + \gamma_{10} R_{t-1}^3 + \gamma_{11} R_{t-2}^3 \end{aligned} \tag{18}$$

Non Linear Multiple Regression Models(NLMRM-6)

$$\begin{aligned} R_t &= \alpha_0 + \alpha_1 T_t + \alpha_2 T_{t-1} + \alpha_3 T_{t-2} + \alpha_4 T_{t-3} + \alpha_5 D_t + \alpha_6 D_{t-1} + \alpha_7 D_{t-2} \\ &\quad + \alpha_8 D_{t-3} + \alpha_9 H_t + \alpha_{10} H_{t-1} + \alpha_{11} H_{t-2} + \alpha_{12} H_{t-3} + \alpha_{13} R_{t-1} + \alpha_{14} R_{t-2} \\ &\quad + \alpha_{15} R_{t-3} + \beta_1 T_t^2 + \beta_2 T_{t-1}^2 + \beta_3 T_{t-2}^2 + \beta_4 T_{t-3}^2 + \beta_5 D_t^2 + \beta_6 D_{t-1}^2 \\ &\quad + \beta_7 D_{t-2}^2 + \beta_8 D_{t-3}^2 + \beta_9 H_t^2 + \beta_{10} H_{t-1}^2 + \beta_{11} H_{t-2}^2 + \beta_{12} H_{t-3}^2 + \beta_{13} R_{t-1}^2 \\ &\quad + \beta_{14} R_{t-2}^2 + \beta_{15} R_{t-3}^2 + \gamma_1 T_t^3 + \gamma_2 T_{t-1}^3 + \gamma_3 T_{t-2}^3 + \gamma_4 T_{t-3}^3 + \gamma_5 D_t^3 \\ &\quad + \gamma_6 D_{t-1}^3 + \gamma_7 D_{t-2}^3 + \gamma_8 D_{t-3}^3 + \gamma_9 H_t^3 + \gamma_{10} H_{t-1}^3 + \gamma_{11} H_{t-2}^3 \\ &\quad + \gamma_{12} H_{t-3}^3 + \gamma_{13} R_{t-1}^3 + \gamma_{15} R_{t-3}^3 \end{aligned} \tag{19}$$

The above three equations model the rainfall intensities using $7, 11$ and 15 operational parameters. The results are discussed and compared in the next chapter. Coefficients obtained in these nonlinear models are summarized in table 4.

Parameters	NLMRM-1	NLMRM-2	NLMRM-3
α_0	0.0043	0.0021	0.0024
α_1	0.0015	-0.0125	0.0013
α_2	-0.0046	0.0071	-0.0023
α_3	0.0015	0.0076	-0.0103
α_4	-0.0039	0.0028	0.0056
α_5	0.0127	-0.0073	-0.0073
α_6	0.0116	0.0089	0.0073
α_7	0.0753	0.1052	-0.0143
α_8		-0.0052	0.0073
α_9		0.0088	0.0029
α_{10}		0.0561	-0.0039
α_{11}		-0.1137	-0.0093
α_{12}			0.0213
α_{13}			-0.0915
α_{14}			-0.0873
α_{15}			0.1163
β_1	-0.0135	0.0321	0.0743
β_2	-0.0164	0.0146	0.0563
β_3	0.0346	-0.0542	0.0854
β_4	0.0485	0.0375	-0.0575
β_5	0.0394	0.0275	0.0473
β_6	-0.1232	0.0637	0.0758
β_7	0.6563	-0.1356	0.1648
β_8		0.0745	0.1562
β_9		0.0864	0.0975
β_{10}		-0.1375	0.0476
β_{11}		0.8543	0.8631
β_{12}			-0.8754
β_{13}			0.3672
β_{14}			-0.9762
β_{15}			0.9871

Table 3: Regression Coefficients of NLMRM-1, NLMRM-2 and NLMRM-3 Models

Parameters	NLMRM-4	NLMRM-5	NLMRM-6
α_0	0.0007	0.0019	0.0057
α_1	0.0122	0.0093	-0.0047
α_2	-0.0049	0.0082	-0.0118
α_3	0.0022	0.0047	0.0073
α_4	-0.0056	0.0071	-0.0019
α_5	0.0042	-0.0061	0.0083
α_6	-0.0092	0.0107	-0.0039
α_7	0.1026	0.0027	-0.0051
α_8		-0.0055	-0.0041
α_9		-0.0143	0.0077
α_{10}		0.0258	-0.0127
α_{11}		-0.0936	-0.0115
α_{12}			0.0143
α_{13}			0.1026
α_{14}			-0.1261
α_{15}			0.0973
β_1	0.0095	-0.0153	-0.0563
β_2	-0.0133	0.0173	0.0554
β_3	0.0167	0.0271	-0.0629
β_4	0.0148	0.0229	0.0389
β_5	-0.0631	0.0172	0.0625
β_6	0.0943	-0.0552	0.0461
β_7	-0.0937	0.1124	-0.0953
β_8		0.0357	0.0936
β_9		0.0864	0.0157
β_{10}		0.1164	-0.1273
β_{11}		0.6931	0.2741
β_{12}			0.7521
β_{13}			-0.1627
β_{14}			0.5734
β_{15}			0.7872
γ_0	0.0146	0.0876	0.1507
γ_1	0.0165	0.0578	0.4523
γ_2	0.0256	0.0845	0.3974
γ_3	0.0457	0.0365	0.5764
γ_4	0.0683	0.0578	0.1673
γ_5	0.0748	0.0837	0.2873
γ_6	0.0865	0.0876	0.4821
γ_7	0.8673	-0.0986	0.3374
γ_8		0.1847	0.2437
γ_9		0.3938	0.1742
γ_{10}		-0.0765	0.2261
γ_{11}		0.9763	0.9643
γ_{12}			-0.2693
γ_{13}			0.6764
γ_{14}			-0.1274
γ_{15}			1.0036

Table 4: Regression Coefficients of NLMRM-4, NLMRM-5 AND NLMRM-6 Models

4.3 Thunderstorm Cell Models

The mechanism underlying air mass lifting and precipitation are illustrated by considering a schematic model of a thunderstorm cell. The thunderstorm is visualized as a vertical column made up of three parts, an *inflow region* near the ground where warm, moist air is drawn into the cell, an *uplift region* in the middle where moisture condenses as air rises, producing precipitation, and an *outflow region* in the upper atmosphere where outflow of cooler, dryer air occurs. The *intensity of precipitation* is given by:

$$i = \frac{4\rho_a V \Delta z}{\rho_w D}\left(\frac{q_{v_1} - q_{v_2}}{1 - q_{v_2}}\right) \tag{20}$$

where i represents intensity of precipitation

ρ_a represents density of moist air

ρ_w represents density of water

V represents wind velocity

Δz represents cloud base or height

D represents diameter of thunderstorm cell

q_{v_1} represents specific humidity at averaged height of cloud

q_{v_2} represents specific humidity at outflow elevation

As the information about the two variables *i.e.* cloud base or height Δz and diameter D were found to be very little or poor, the idea of calibrating it made it compulsory to develop the two models which are discussed briefly in following sections. The ratio cloud base to diameter of cell diameter $\Delta z/D$ is assumed to be α for easiness.

4.3.1 Thunderstorm Cell Models (TCM-1)

In this model the ratio of cloud base to diameter of cell diameter α is assumed to be unity and calculations were performed according to above equation. The results thus obtained were not found to be very good as discussed in next chapter. The incapability of getting reasonable results was the motivation for developing the other Thunderstorm Cell model.

4.3.2 Thunderstorm Cell Models (TCM-2)

In this model the ratio of cloud base to diameter of cell diameter α were calibrated using the training set of data and then used for calculating intensity of precipitation after averaging the value of above said ratio. Available data were used to calibrate the TCM-2 model for finding the above said parameter α. The calculation is made

using the TCM equation (19). The results this way found were better as compared to TCM-1 model. The results are discussed in next chapter in detail.

4.4 Persistence Method

The Persistence Method is the simplest way of producing a forecast. The persistence method assumes that the conditions at the time of the forecast will not change. For example, if two inches of rain fell at any time interval the persistence method would predict two inches of rain for next time interval. The persistence method works well in places where weather conditions ,vary little from day to day. However, if weather conditions change significantly from day to day, the persistence method usually breaks down and is not the best forecasting method to use.

It may also appear that the persistence method would work only for shorter-term forecasts (e.g. a forecast for a day or two), but actually one of the most useful roles of the persistence forecast is predicting long range weather conditions or making climate forecasts. For example, it is often the case that one rainy month will be followed by another rainy month. So, making persistence forecasts for monthly and seasonal weather conditions can have some skill. It may be presented by following equation:

$$R_t = R_{t-1} \tag{21}$$

where R_t represents rainfall forecasted at time-step t and

R_{t-1} represents rainfall observed at time-step $t-1$

The Results obtained from this method also were not found to be very reasonable and are discussed in next chapter.

Chapter 5

Results and Discussions

In this study, three types of models have been developed for forecasting hourly rainfall. Persistence method is also implemented to forecast the rainfall intensity. All the models and methods are discussed in the previous chapter in detail. The whole dataset was divided in two parts; one consisting of 166 data points was used for training/calibration of various models while the other containing 46 data points was used for testing purposes to evaluate the performance of various models developed in this study. Before discussing the results obtained from various models in this study, let us look at various statistical parameters to evaluate the performance of various models.

5.1 Statistical Parameters

To evaluate the performance of various models four different types of statistical parameters are used so as to make an easy and quick comparison of all models. The parameters are discussed in following subsections.

5.1.1 Average Absolute Relative Error (AARE)

Average absolute relative error (AARE) is the average of all the absolute values of errors affiliated with each data points in forecasting an event. The AARE may be computed only after relative error of each particular variable corresponding to it's

exact value is known. Relative error (RE_t) can be computed as

$$RE_{(t)} = \left(\frac{RO_{(t)} - RF_{(t)}}{RO_{(t)}} \right) X100 \tag{22}$$

where $RE_{(t)}$ = Relative error in forecasting $RO_{(t)}$

$RO_{(t)}$ = Observed Rainfall at time 't'

$RF_{(t)}$ = Forecasted rainfall at time 't'

Relative error may be positive or negative. While calculating AARE absolute values of relative errors are considered. AARE may be computed as follows:

$$AARE = \frac{1}{N} \sum_{t=1}^{N} |RE_{(t)}| \tag{23}$$

where N = total number of data points forecasted

It is obvious from above definition, lower the value of AARE, better is the performance of model or vice-versa.

5.1.2 Threshold Statistics

Threshold statistics, a measure of the model performance, is defined for a certain level of relative error, let's say $x\%$. It gives the percentage of the data points predicted for which the relative error is less than or equal to desired level. Mathematically,

$$TS_{(x)} = 100 \frac{n}{N} \% \tag{24}$$

where n = the number of data points whose relative error is less than $x\%$

N = Total number of data points forecasted

As obvious from its definition, higher the value of threshold statistics, better is the performance of model or vice-versa.

5.1.3 Correlation Coefficient

Correlation coefficient is the appropriate measure of relationship between the two variables, forecasted and observed rainfall intensities in the present study. Correlation coefficient (R^2) can not exceed unity numerically. It always lies between +1 and −1. Correlation is said to be perfect and positive if R^2 equals +1 and negative if R^2 equals −1. It is clear that model performance is proportional to the correlation coefficient. A convenient form of the formula for computational work is as follows:

$$R^2 = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\left[\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2 \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2\right]^{1/2}} \tag{25}$$

where $(x_i - \bar{x})$ is deviation of observed value from its mean

$(y_i - \bar{y})$ is deviation of calculated value from its mean

5.1.4 Standard Deviation

Standard deviation usually denoted by σ is the positive square root of the arithmetic mean of the squares of the deviations of the given values from their arithmetic mean. The values in use here are forecasted precipitation intensities. It is computed as follows:

$$\sigma = \sqrt{\frac{1}{N}\sum f_i(x_i - \bar{x})^2} \tag{26}$$

where $\bar{x}$ is the arithmetic mean and $\sum f_i = N$

Statistical performance of various optimum models in training and testing are tabulated in Table 5 and 6 respectively. Averaged statistical performance of optimum ANN models, Regression models, TCM models are tabulated further in Table 7. Observed and forecasted rainfall intensities from various models for training as well as testing are shown at the end of this chapter.

MODEL	AARE	CORR.COEFF	STD. DEV.	TS1	TS5	TS10	TS25	TS50	TS75	TS100
7-13-1 ANN	17.80	0.9961	0.0127	8.49	24.52	43.86	76.41	92.92	98.11	99.05
11-18-1 ANN	14.41	0.9977	0.0099	9.90	30.18	46.69	82.54	97.16	99.52	100.00
15-18-1 ANN	14.46	0.9980	0.0089	12.26	34.43	53.77	81.13	95.28	98.58	100.00
7-10-5-1 ANN	10.01	0.9983	0.0089	16.03	38.67	56.60	80.18	97.16	100.00	100.00
11-13-5-1 ANN	13.71	0.9983	0.0086	16.04	38.67	52.83	80.66	97.64	100.00	100.00
15-11-8-1 ANN	9.64	0.9987	0.0079	17.45	39.15	57.07	84.46	99.02	100.00	100.00
TCM-1	724.45	0.1537	0.0325	0.00	0.47	0.97	0.97	1.94	9.22	23.78
TCM-2	69.16	0.3956	0.0305	0.97	4.85	12.62	28.15	51.45	65.53	89.80
LMRM-1	118.05	0.4932	0.0672	0.97	1.94	7.28	16.50	38.34	51.45	61.65
LMRM-2	110.94	0.5235	0.0531	0.00	1.45	7.28	21.35	40.77	55.82	64.56
LMRM-3	103.24	0.5434	0.0431	1.94	4.36	10.67	25.72	44.66	60.19	66.01
NLMRM-1	87.94	0.7428	0.0451	0.48	10.19	18.93	31.06	46.11	63.10	72.81
NLMRM-2	74.59	0.7129	0.0446	3.88	9.22	14.56	30.58	51.45	69.90	77.66
NLMRM-3	62.52	0.7435	0.0402	0.48	8.25	14.07	32.03	56.79	74.27	82.52
NLMRM-4	60.77	0.7453	0.0402	1.94	7.28	22.33	43.20	64.56	76.69	83.00
NLMRM-5	56.15	0.7735	0.0398	3.88	10.67	20.38	39.32	62.62	75.72	82.52
NLMRM-6	44.32	0.7947	0.0356	2.42	15.53	27.18	53.39	74.75	82.52	86.89
Persistence Model	302.24	0.7965	0.0473	7.83	10.84	12.65	16.86	31.92	45.78	52.40

Table 5: Statistical Parameters of Optimum Models in Training

Model	AARE	Corr. Coeff.	Std. Dev.	TS1	TS5	TS10	TS25	TS50	TS75	TS100
7-13-1 ANN	19.63	0.9952	0.0138	6.13	24.05	40.09	73.58	91.98	96.69	99.05
11-18-1 ANN	18.12	0.9966	0.0115	8.01	26.41	43.86	76.88	93.39	96.69	98.58
15-18-1 ANN	19.14	0.9970	0.0110	12.26	28.30	45.75	73.58	91.98	96.69	98.58
7-10-5-1 ANN	14.41	0.9977	0.0099	9.90	30.18	46.69	82.54	97.16	99.52	100.00
11-13-5-1 ANN	18.34	0.9969	0.0119	11.79	25.00	44.33	69.33	93.39	99.52	100.00
15-11-8-1 ANN	13.41	0.9984	0.0084	17.45	39.15	55.66	80.18	97.16	100.00	100.00
TCM-1	840.29	0.1325	0.0345	0.00	0.47	0.47	4.72	9.43	11.32	16.51
TCM-2	78.27	0.3256	0.0365	0.00	4.76	9.52	16.67	19.05	42.86	95.24
LMRM-1	144.91	0.4653	0.0704	0.00	0.48	3.88	6.31	17.96	41.26	56.31
LMRM-2	121.12	0.4895	0.0593	0.00	1.45	6.79	16.01	32.52	48.54	60.19
LMRM-3	109.78	0.5194	0.0507	0.00	1.45	10.19	15.04	34.46	50.00	60.67
NLMRM-1	95.40	0.6744	0.0492	0.48	5.82	15.04	33.00	50.48	59.22	67.96
NLMRM-2	82.69	0.6970	0.0473	0.97	11.16	18.44	41.26	53.39	66.50	76.21
NLMRM-3	68.83	0.7129	0.0429	1.94	9.22	18.44	40.29	59.70	71.84	77.66
NLMRM-4	64.88	0.7255	0.0436	2.42	7.28	16.99	34.46	59.22	75.24	79.12
NLMRM-5	59.73	0.7284	0.0422	2.91	10.19	19.90	40.77	62.62	73.30	83.49
NMLRM-6	46.58	0.7746	0.0376	3.39	14.56	25.24	49.02	71.35	79.61	85.43
Persistence Method	246.39	0.8012	0.0412	4.35	4.35	6.52	10.87	34.78	41.30	52.17

Table 6: Statistical Parameters of Optimum Models in Testing

Model	AARE	Corr. Coeff.	Std. Dev.	TS1	TS5	TS10	TS25	TS50	TS75	TS100
Training										
ANN (Three-layer)	15.55	0.9973	0.0671	10.22	29.71	48.11	80.03	95.12	98.74	99.68
ANN(Four-layer)	11.12	0.9984	0.8530	16.50	38.83	55.50	84.43	97.94	100.00	100.00
TCM-2	69.16	0.3956	0.0305	0.97	4.85	12.62	28.15	51.45	65.53	89.80
LMRM	110.74	0.5203	0.5451	0.97	2.58	8.41	21.19	41.26	55.82	64.07
NLMRM	64.38	0.7521	0.0409	2.18	10.19	19.57	38.27	59.38	73.70	80.91
Persistence Model	302.24	0.7965	0.0473	7.83	10.84	12.65	16.86	31.92	45.78	52.40
Testing										
ANN(Three-layer)	18.97	0.9963	0.0121	8.80	26.25	43.23	74.68	92.45	96.69	98.74
ANN(Four-layer)	15.39	0.9977	0.0098	13.05	31.44	48.89	77.35	95.90	98.68	100.00
TCM-2	78.27	0.3256	0.0365	0.00	4.76	9.52	16.67	19.05	42.86	95.24
LMRM	125.27	0.4914	0.0601	0.00	1.13	6.95	12.45	17.97	46.60	59.06
NLMRM	69.68	0.7188	0.0439	2.02	9.70	19.01	39.80	59.46	70.95	78.31
Persistence Method	246.39	0.8012	0.0412	4.35	4.35	6.52	10.87	34.78	41.30	52.17

Table 7: Averaged Statistical Parameters of Optimum Models in Training and Testing

5.2 Discussions of Results

The results tabulated in Table 5, 6 and 7 are self explanatory. Discussion about training and testing sets are made distinctively. The discussion of results have been divided into three parts as follows:

5.2.1 Results During Training

Taking into consideration the training set models it is found that the least AARE of 9.65% was obtained from 15-11-8-1 ANN model whereas TCM-1 model was showing largest AARE of 729.45%. The NLMRM-6 model captured AARE of 44.32% and Persistence method obtained an AARE of 302.25%. So based on the results in terms of AARE during training, it can be concluded that the 15-11-8-1 ANN models learnt better than other models.

The best correlation was also shown by the same ANN model which gave the coefficient value of 0.9987 followed by Persistence method with a value of 0.7965, whereas TCM-1 model was found to be giving the correlation coefficient of 0.1537. So based on the results in terms of correlation coefficient during training, it can be concluded that the 15-11-8-1 ANN models learnt better than other models.

The TS1 for 15-11-8-1 ANN model was found to be 17.45% whereas 7-10-5-1 and 11-13-5-1 ANN models had 16.04%. Persistence model followed the ANN models with a value of 7.83% for TS1. There was no observation having relative error in forecasting less than 1% from TCM-1 and LMRM-2 models. The TS10 value obtained through 15-11-8-1 ANN model was found to be 57.07% followed by 7-10-5-1 ANN model with a close value of 56.60%. NLMRM-5 and NLMRM-4 obtained the TS10 values as 27.18% and 22.33% respectively. TS50 value was found 99.03% for 15-11-8-1 ANN model followed by 11-13-5-1, 7-10-5-1 and 11-18-1 ANN models with the values 97.64%, 97.16% and 97.16% respectively. It was observed that all the forecasted values were having relative error less than 75% for 15-11-8-1, 11-13-5-1 and 7-10-5-1 ANN models whereas TCM-1 was having the only 9.22% forecasted values

less than or equal to 75%. All the threshold statistics were found to be best in case of 15-11-8-1 ANN model whereas TCM-1 model was showing worst threshold statistics.

Hence we may conclude that 15-11-8-1 ANN model had learnt well and was most suited to forecast the rainfall intensity as judged by the statistical performance.

5.2.2 Results During Testing

Similarly, for the testing set also the 15-11-8-1 ANN model was found to give the best results in respect of statistical parameters. AARE of 13.41% was obtained from 15-11-8-1 ANN model, whereas TCM-1 model was showing largest AARE of 840.29%. The NLMRM-6 model captured AARE of 46.58% and Persistence method obtained an AARE of 246.39%. So based on the results in terms of AARE during training, it can be concluded that the 15-11-8-1 ANN models learnt better than other models.

The best correlation was also shown by the same ANN model which gave the coefficient value of 0.9984 followed by Persistence method with a value of 0.8012, whereas TCM-1 model and NLMRM-6 models obtained 0.1324 and 0.7746 values respectively. So based on the results in terms of correlation coefficient during training, it can be concluded that the 15-11-8-1 ANN models learnt better than other models.

The TS1 for 15-11-8-1 ANN model was found to be 17.45% whereas 7-10-5-1 and 11-13-5-1 models had 9.90% and 11.79%. Persistence model followed the ANN models with a value of 4.35% for TS1. There was no observation having relative error in forecasting less than 1% from TCM and LMRM models. The TS10 value obtained through 15-11-8-1 ANN model was found to be 55.66% followed by 7-10-5-1 ANN model with a close value of 46.69%. NLMRM-6 and NLMRM-5 obtained the TS10 values as 25.24% and 19.90% respectively. TS50 value was found 97.16% for 15-11-8-1 and 7-10-5-1 ANN models followed by 11-13-5-1 and 11-18-1 ANN models with ANN models with the values 93.39% and 93.39% respectively. The largest value of TS100 of 100.00% was found from 15-11-8-1, 11-13-5-1 and 7-10-5-1 ANN models whereas TCM-1 was giving the lowest values of 16.51%. TCM-2 obtained

the TS100 value of 95.24% and NLMRM-6 was showing the value of 85.43%. Overall the threshold statistics were found to be best in case of 15-11-8-1 ANN model whereas TCM-1 model was showing worst threshold statistics.

Hence we may conclude that 15-11-8-1 ANN model had learnt well and was most suited to forecast the rainfall intensity as judged by the statistical performance.

5.2.3 Comparisons Among Various Techniques

Statistical parameters were also evaluated on averaged values of all the models in training as well as in testing. AARE for ANN (three-layer) in training was found to be equal to 15.55% whereas ANN (four-layer) obtained a value of 11.12%, TCM-2 obtained a value of 69.16% and NLMRM had a low value of 64.38%. Persistence had an AARE of 302.24%. So based on the results in terms of AARE during training, it can be concluded that the ANN (four-layer) models learnt better than other models.

The best correlation was also shown by the same ANN four-layer model which gave the coefficient value of 0.9984 followed by Persistence method with a value of 0.7965, whereas TCM-2 model and NLMRM models obtained 0.3956 and 0.7521 values respectively. So based on the results in terms of correlation coefficient during training, it can be concluded that the ANN four-layer models learnt better than other models.

Similarly TS1 value was found to be 16.50% for ANN (four-layer) while ANN (three-layer), TCM, LMRM and NLMRM obtained 10.22%, 0.48% , 0.97% and 2.18%. TS10 value was found to be 55.50% for ANN (four-layer) while ANN (three-layer), TCM, LMRM and NLMRM obtained 48.11%, 6.79% , 8.41% and 19.57%. Similarly TS50 value was found to be 97.94% for ANN (four-layer) while ANN (three-layer), TCM, LMRM and NLMRM obtained 95.12%, 26.69% , 41.26% and 59.38%. TS100 value was found to be 100.00% for ANN (four-layer) while ANN (three-layer), TCM, LMRM and NLMRM obtained 99.6855%, 56.79% , 64.07% and

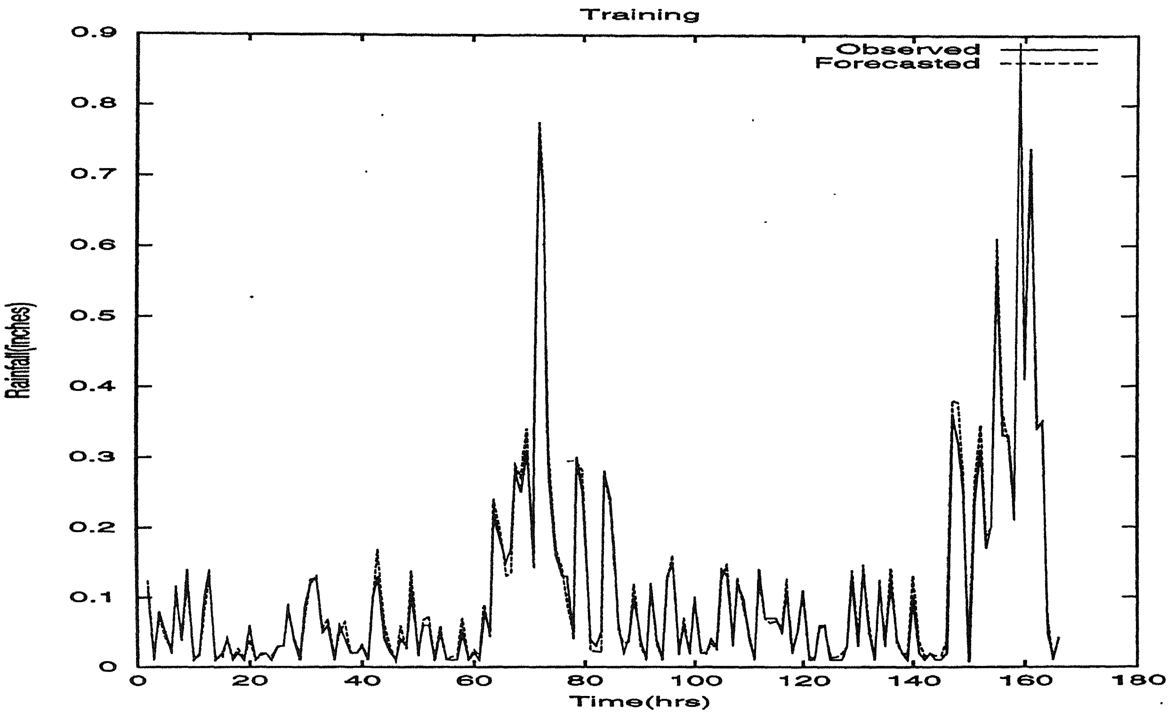
80.91%. Overall the threshold statistics were found to be best in case of ANN four-layer models whereas Persistence model was showing worst threshold statistics.

For the testing set also AARE for ANN (three-layer)was found to be equal to 18.97% whereas ANN (four-layer) obtained a value of 15.39%, TCM-2 obtained a value of 78.27% and NLMRM had the lowest value of 69.68%. Again ANN four-layer models were found most suited for the concerned study.

The best correlation was also shown by the same ANN four-layer model which gave the coefficient value of 0.9963 followed by Persistence method with a value of 0.8012, whereas TCM-2 model and NLMRM models obtained 0.3256 and 0.7188 values respectively. So based on the results in terms of correlation coefficient during training, it can be concluded that the ANN four-layer models learnt better than other models.

Similarly TS1 value was found to be 13.05% for ANN (four-layer) while ANN (three-layer), TCM, LMRM and NLMRM obtained 8.80%, 2.61% , 1.13% and 9.70%. TS10 value was found to be 48.89% for ANN (four-layer) while ANN (three-layer), TCM, LMRM and NLMRM obtained 43.23%, 4.99% , 6.95% and 19.01%. Similarly TS50 value was found to be 95.90% for ANN (four-layer) while ANN (three-layer), TCM, LMRM and NLMRM obtained 92.45%, 14.24% , 17.97% and 59.46%. TS100 value was found to be 100.00% for ANN (four-layer) while ANN (three-layer), TCM, LMRM and NLMRM obtained 98.74%, 55.87% , 59.06% and 78.32%. Overall the threshold statistics were found to be best in case of ANN four-layer models whereas Persistence model was showing worst threshold statistics.

Overall it was observed that the ANN (four-layer) model produced best results as far as statistical parameters are concerned.



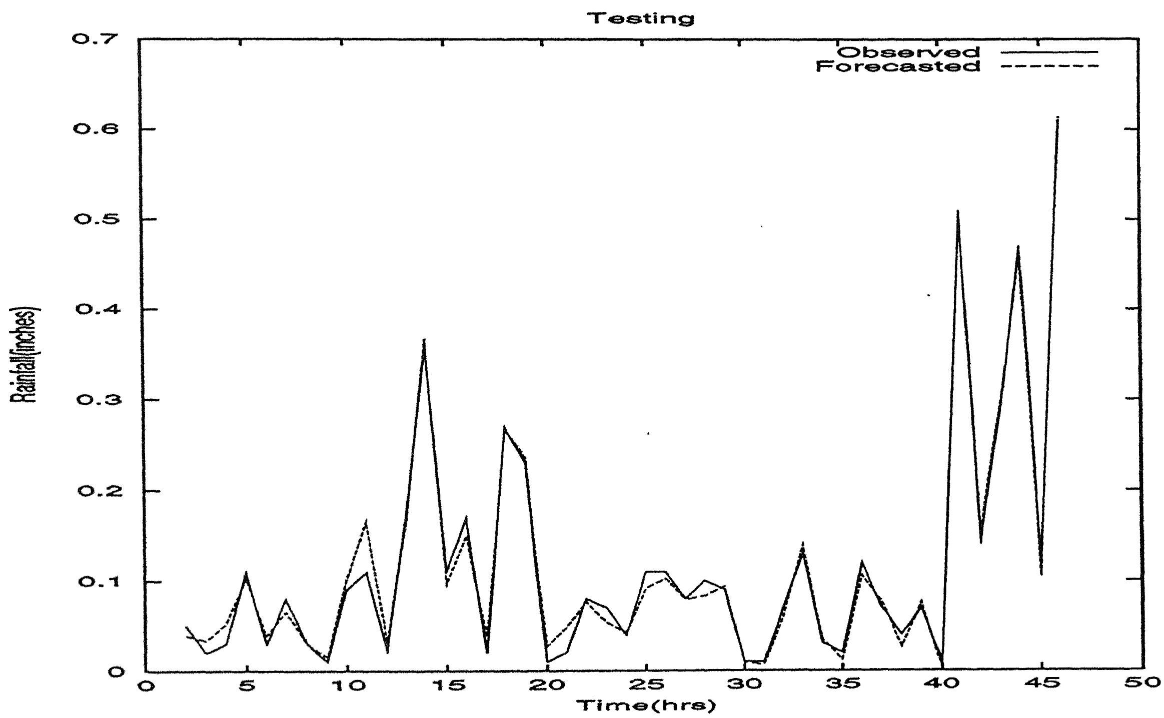


Figure 7: Observed and Forecasted Rainfalls from 7-13-1 ANN Model

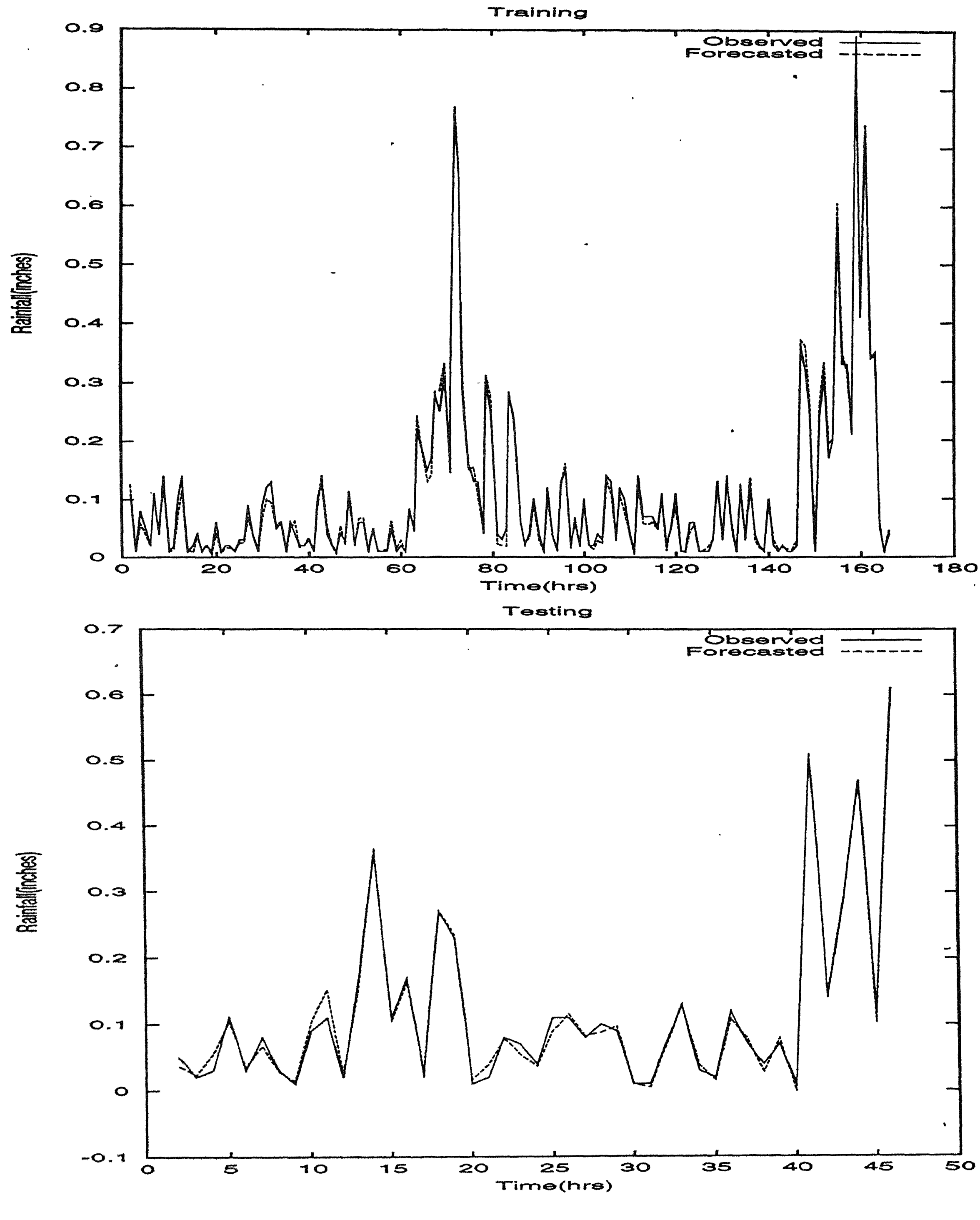
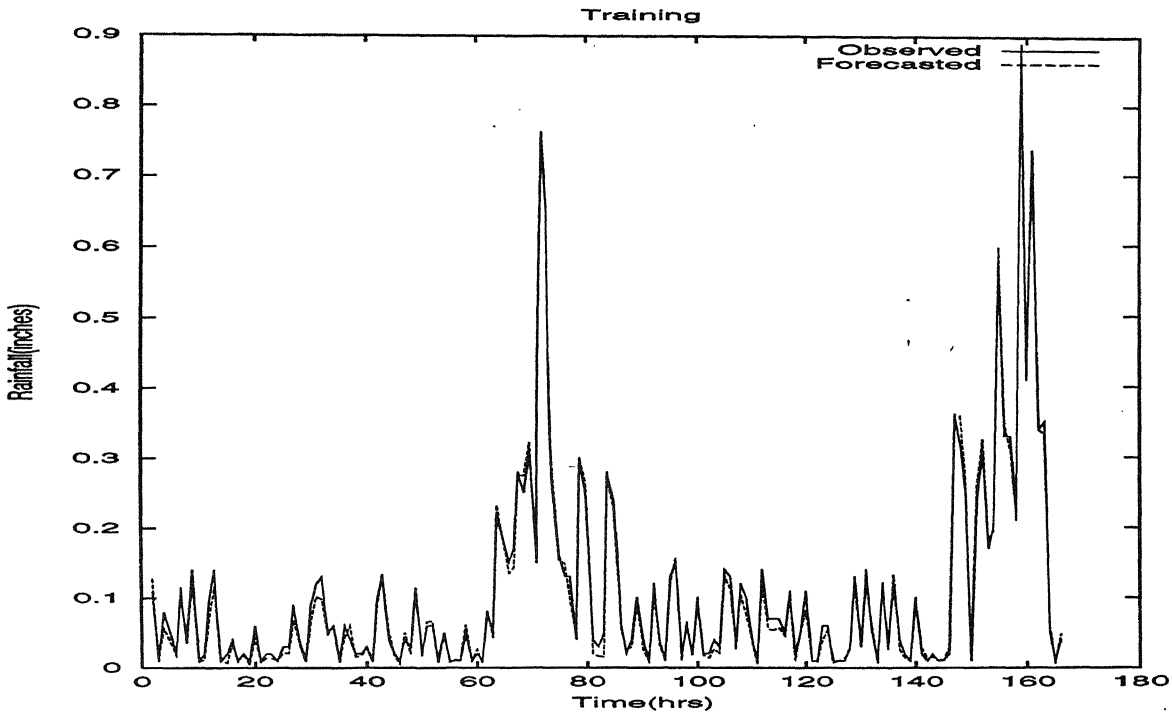


Figure 8: Observed and Forecasted Rainfalls from 11-18-1 ANN Model



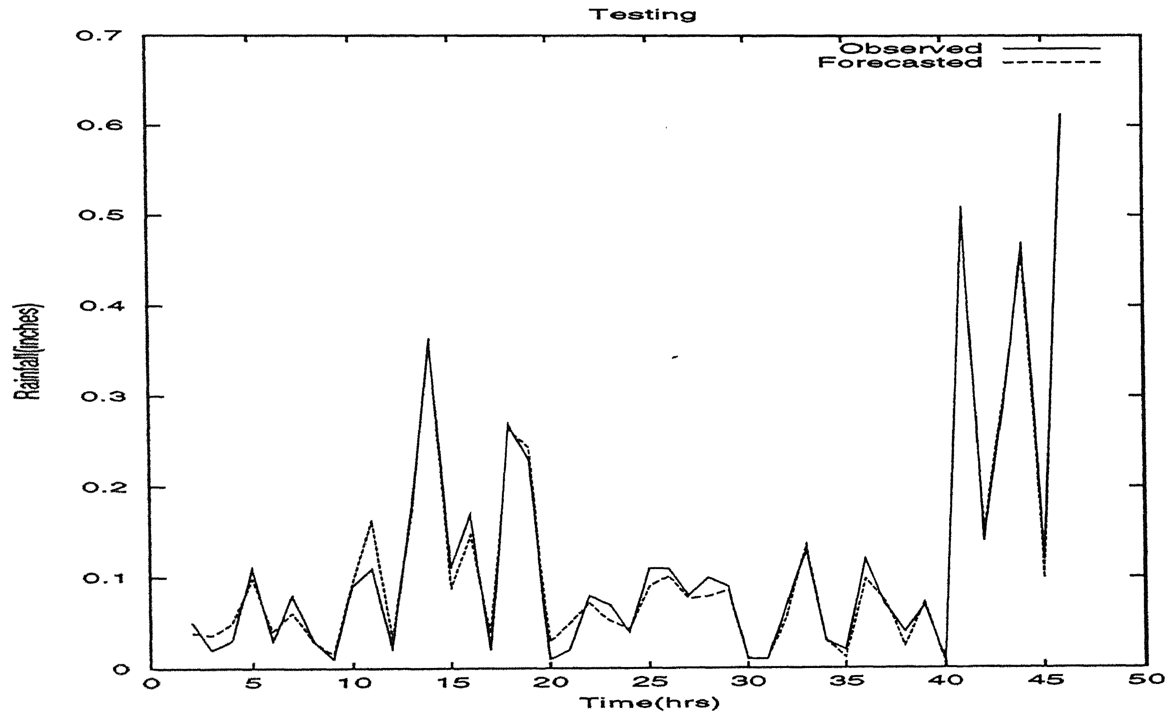
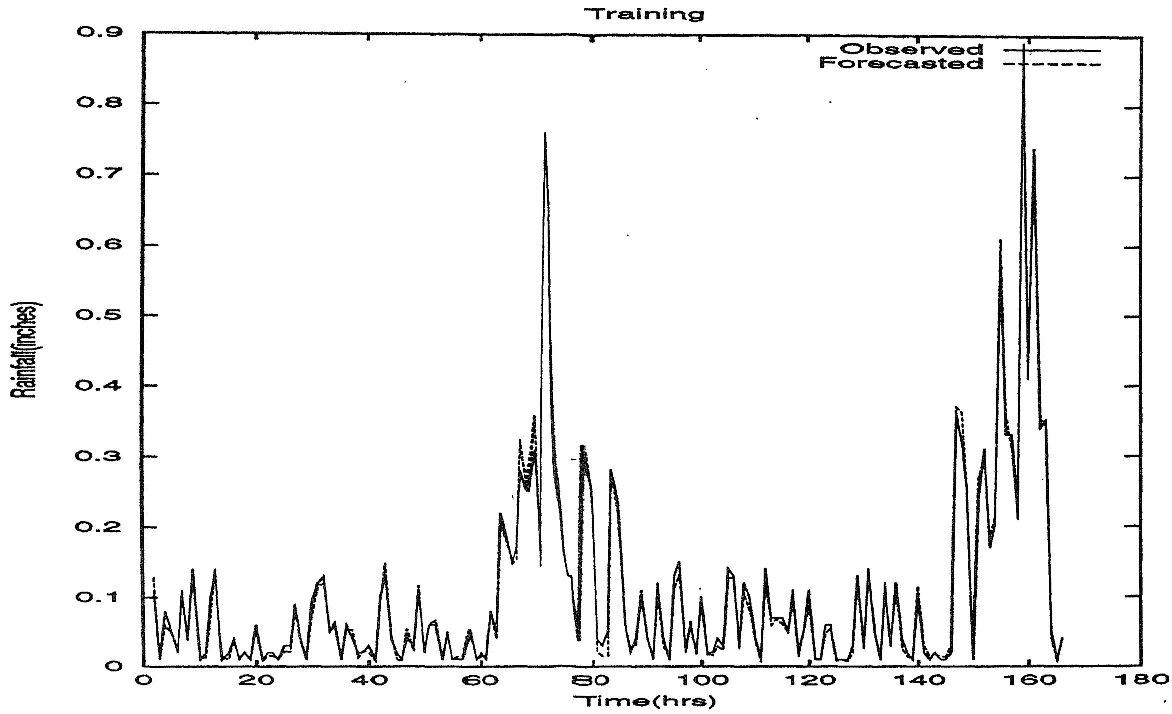


Figure 9: Observed and Forecasted Rainfalls from 15-18-1 ANN Model



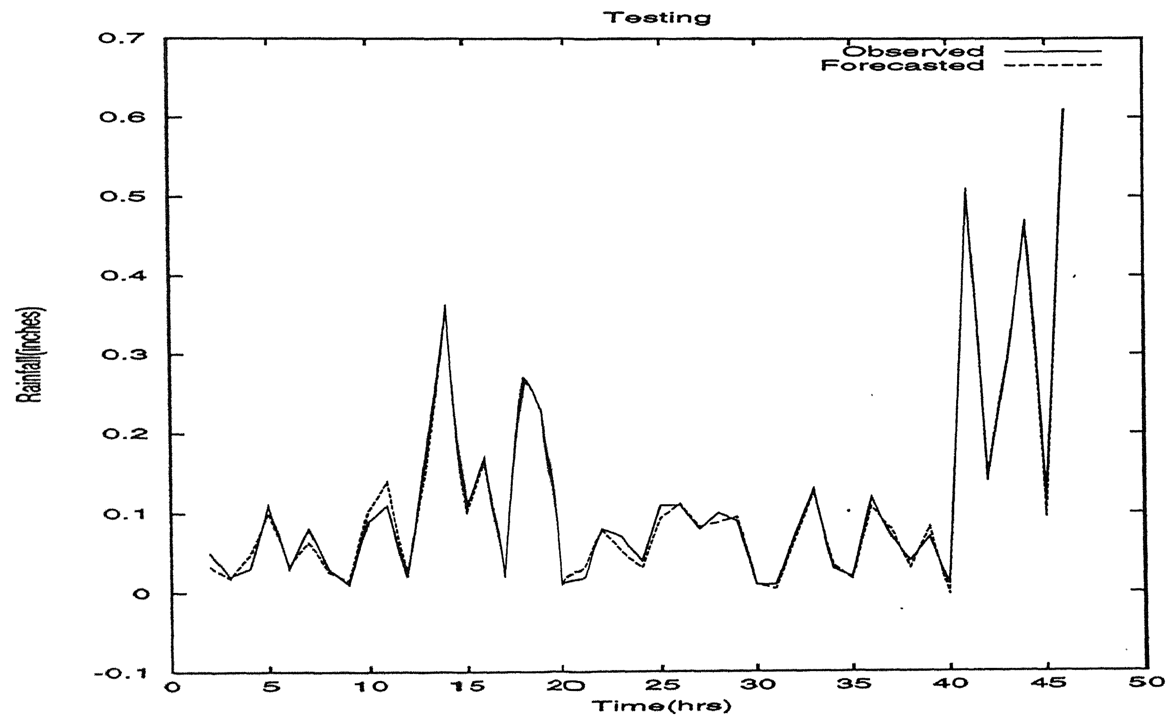


Figure 10: Observed and Forecasted Rainfalls from 7-10-5-1 ANN Model

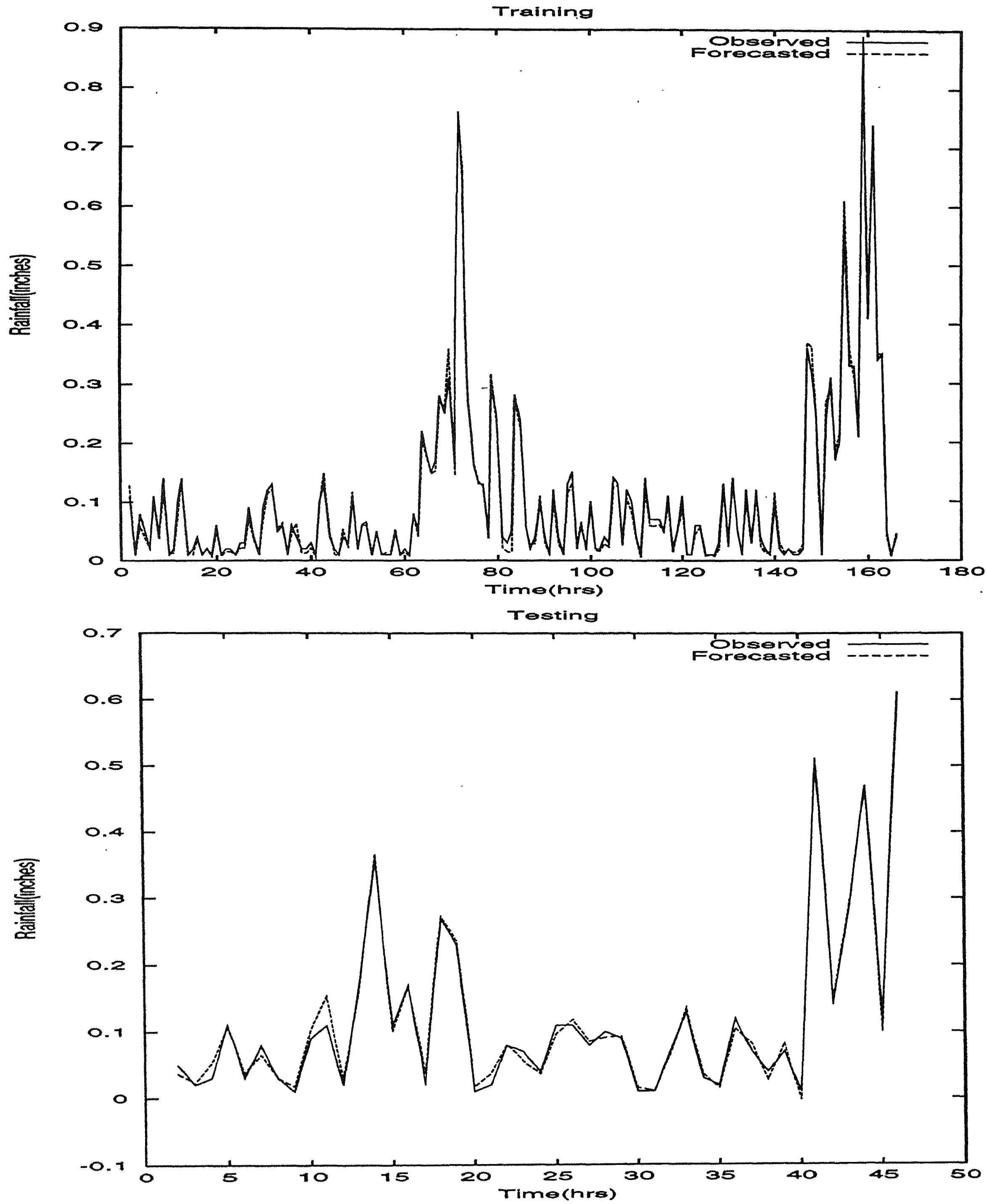


Figure 11: Observed and Forecasted Rainfalls from 11-13-5-1 ANN Model

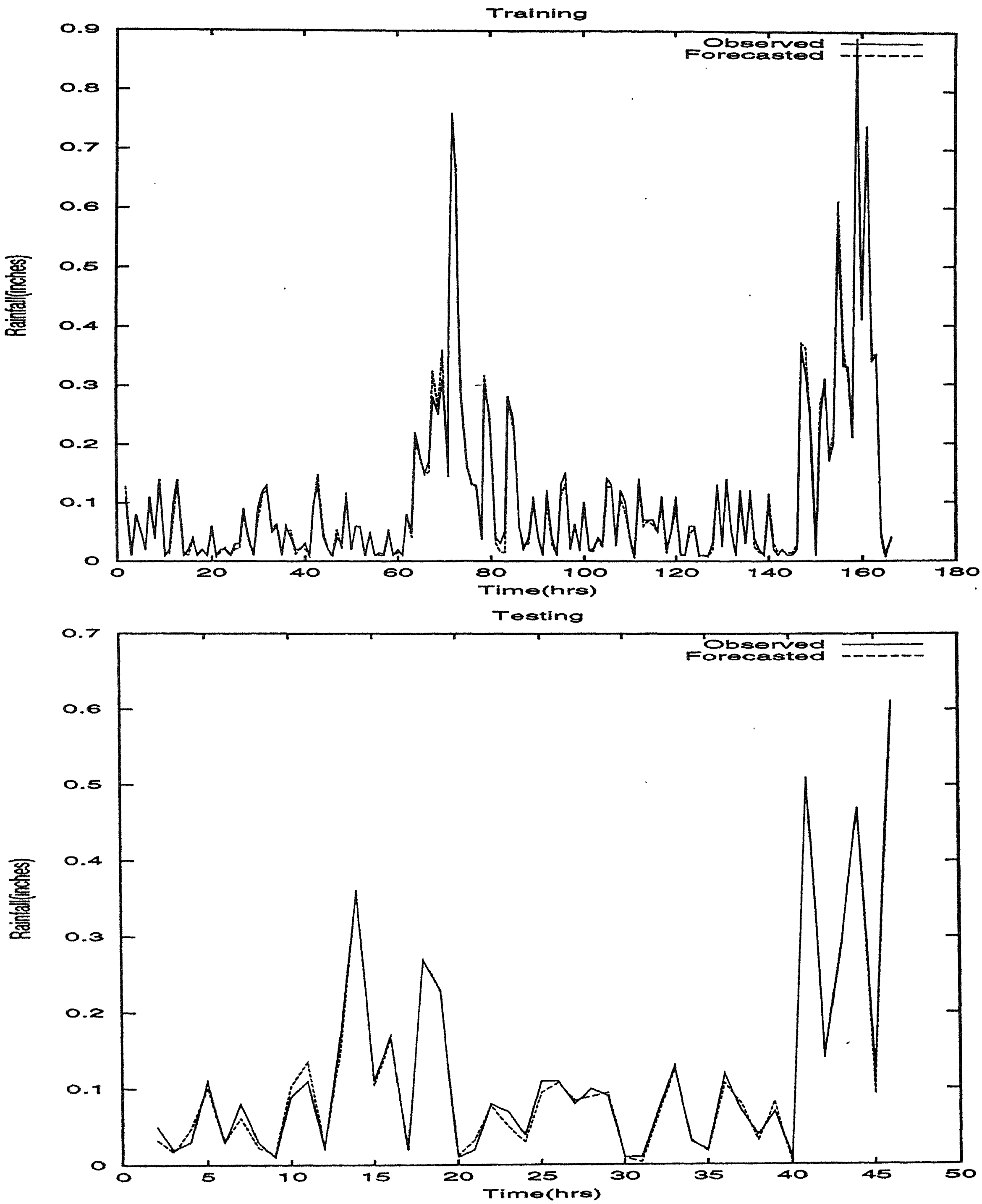


Figure 12: Observed and Forecasted Rainfalls from 15-11-8-1 ANN Model

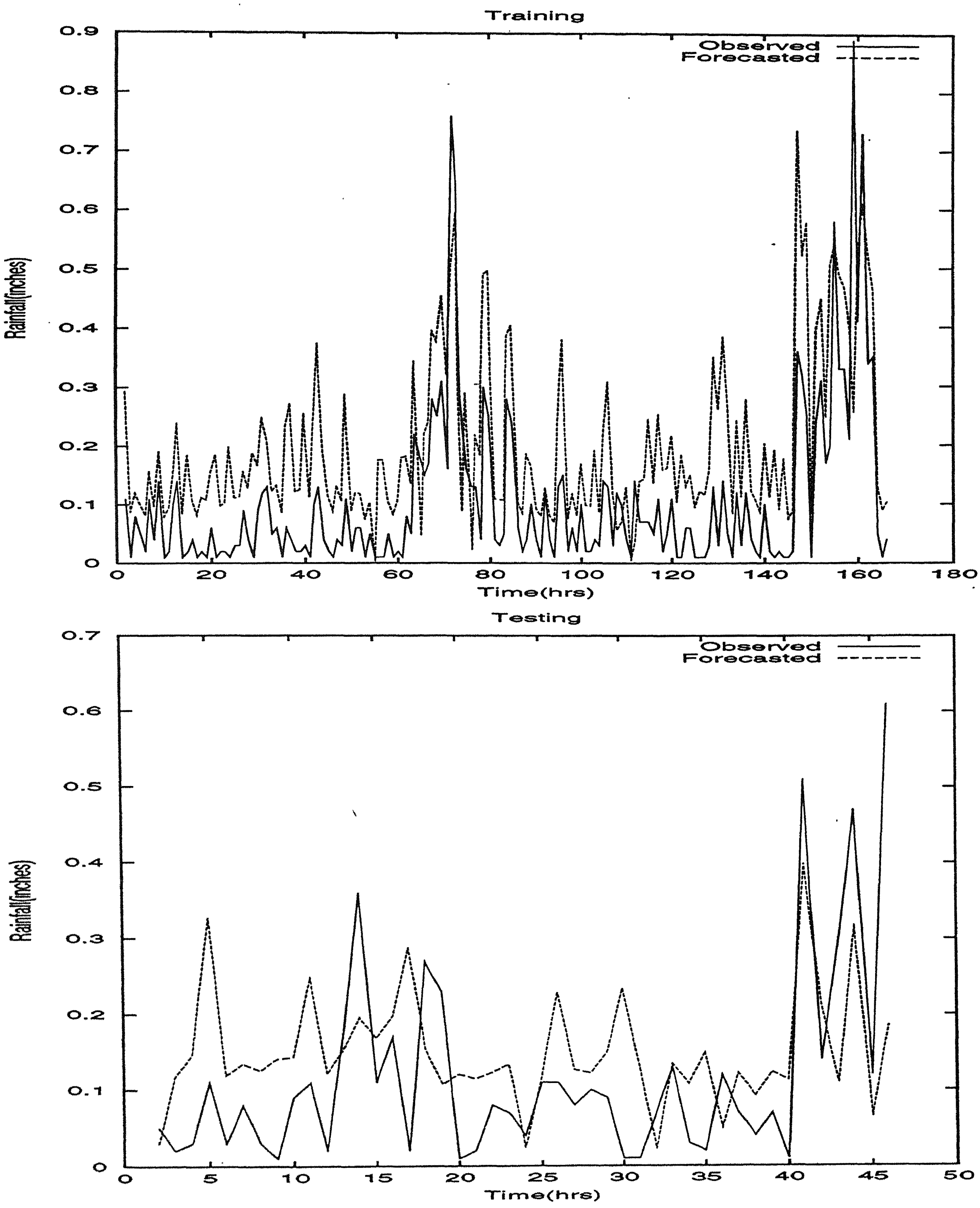


Figure 13: Observed and Forecasted Rainfalls from TCM-1 Model

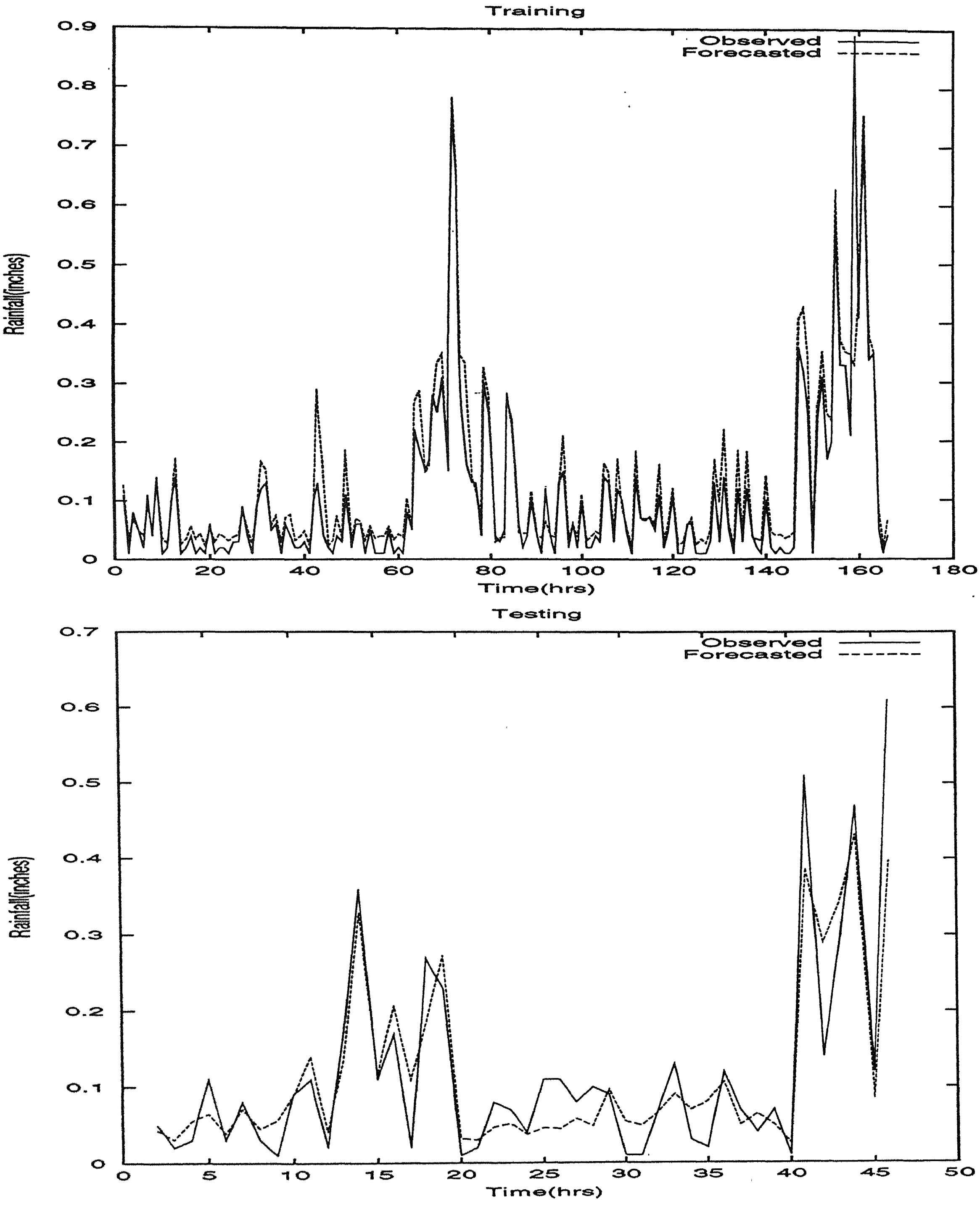


Figure 14: Observed and Forecasted Rainfalls from TCM-2 Model

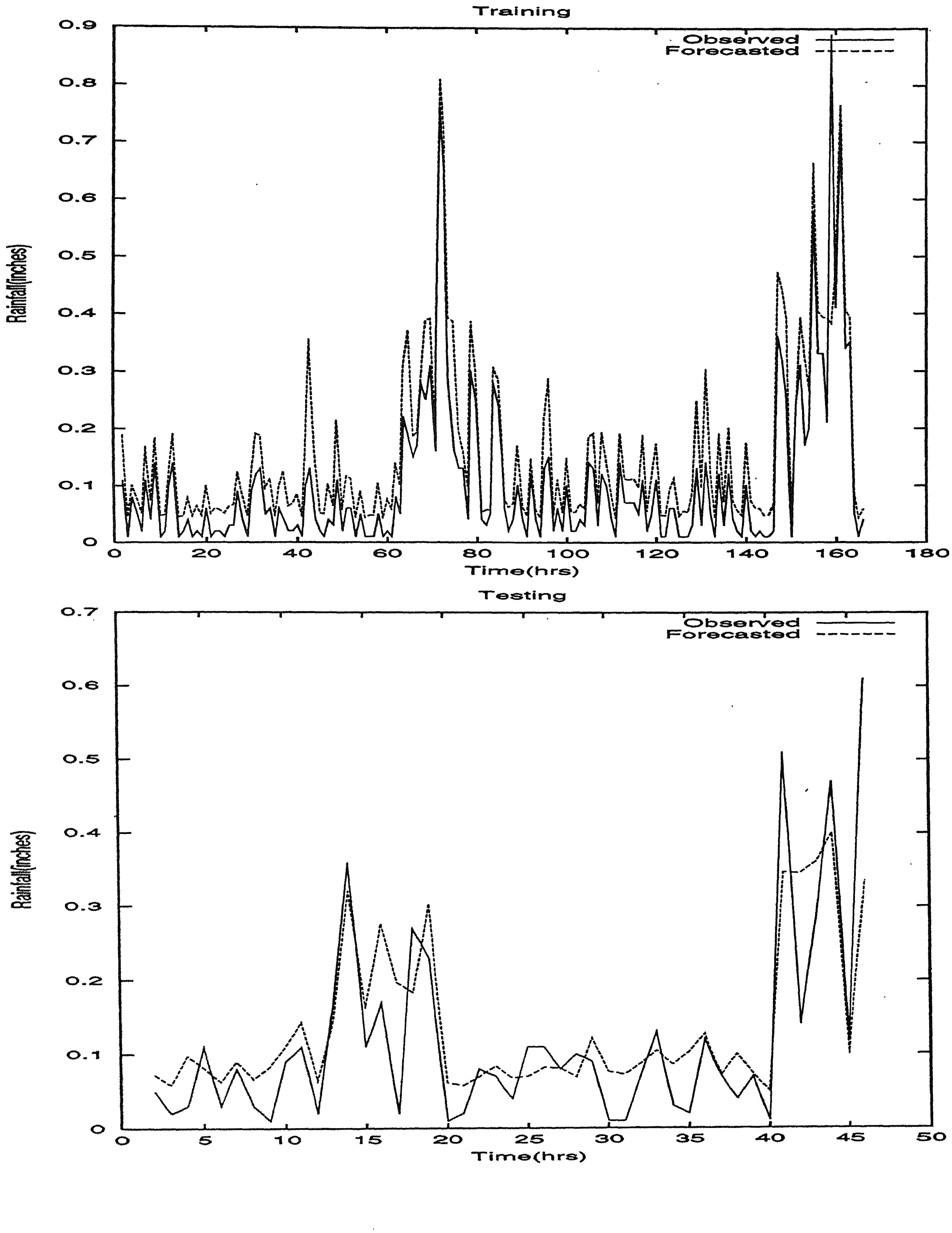


Figure 15: Observed and Forecasted Rainfalls from LMRM-1 Model

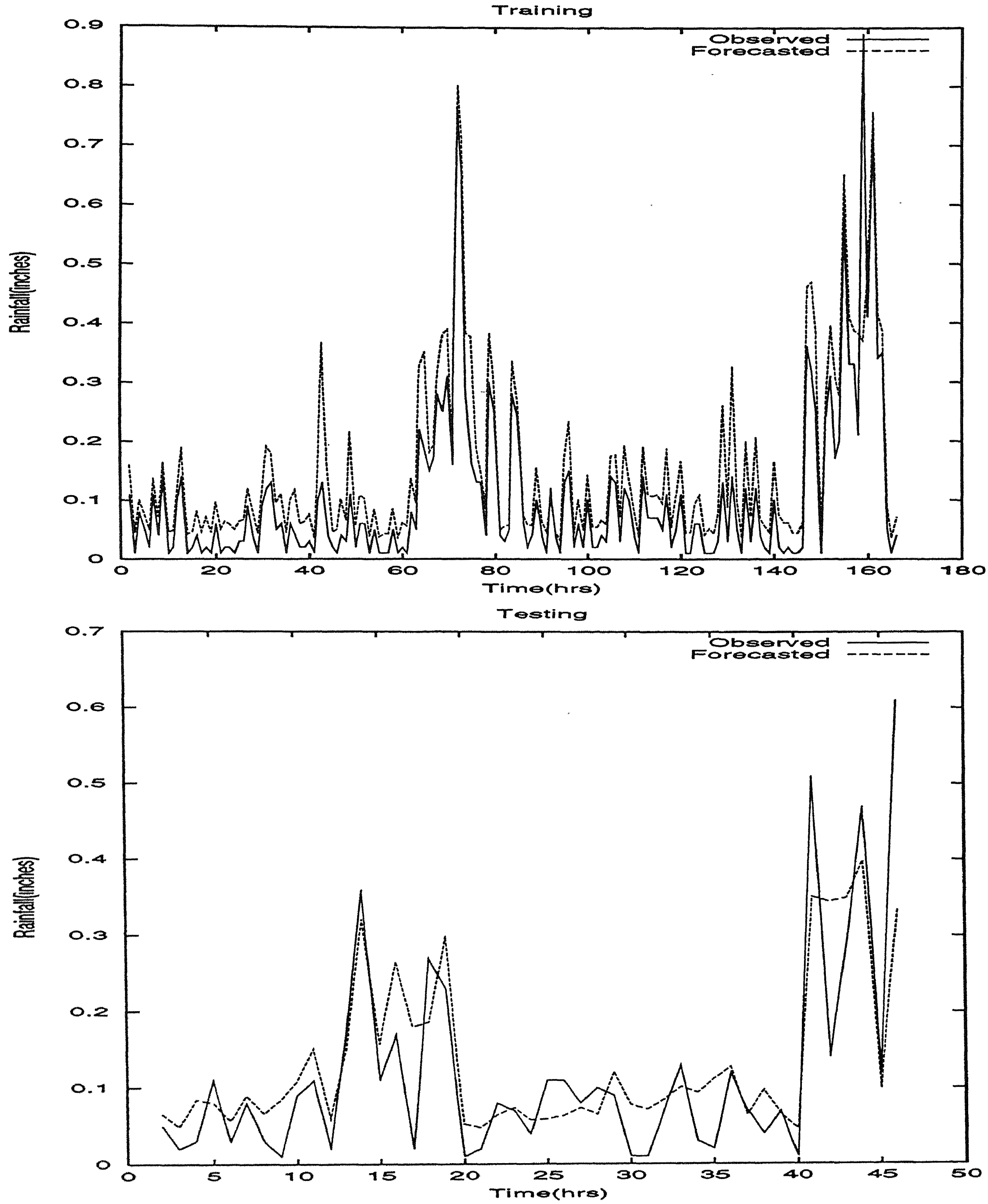


Figure 16: Observed and Forecasted Rainfalls from LMRM-2 Model

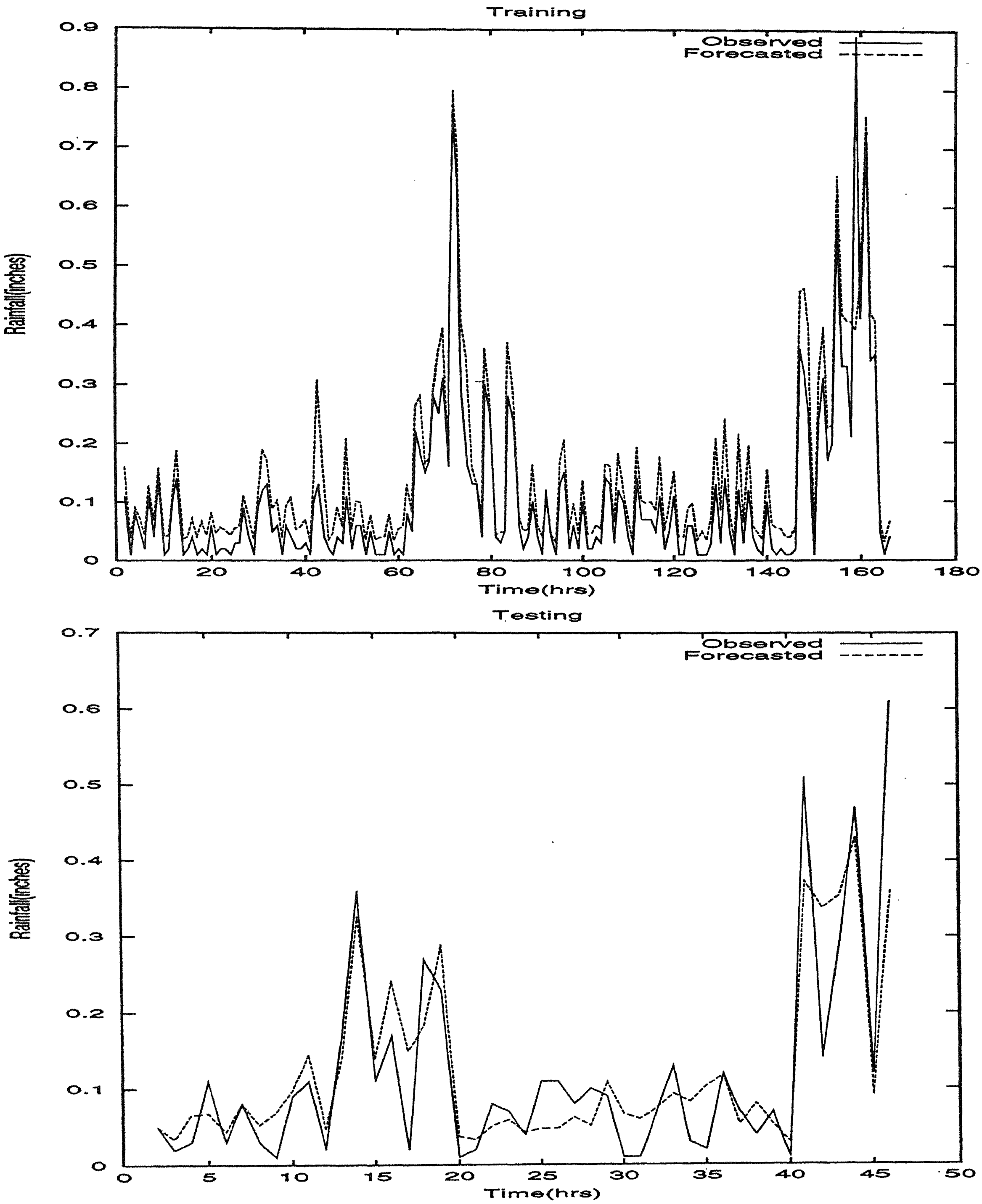


Figure 17: Observed and Forecasted Rainfalls from LMRM-3 Model

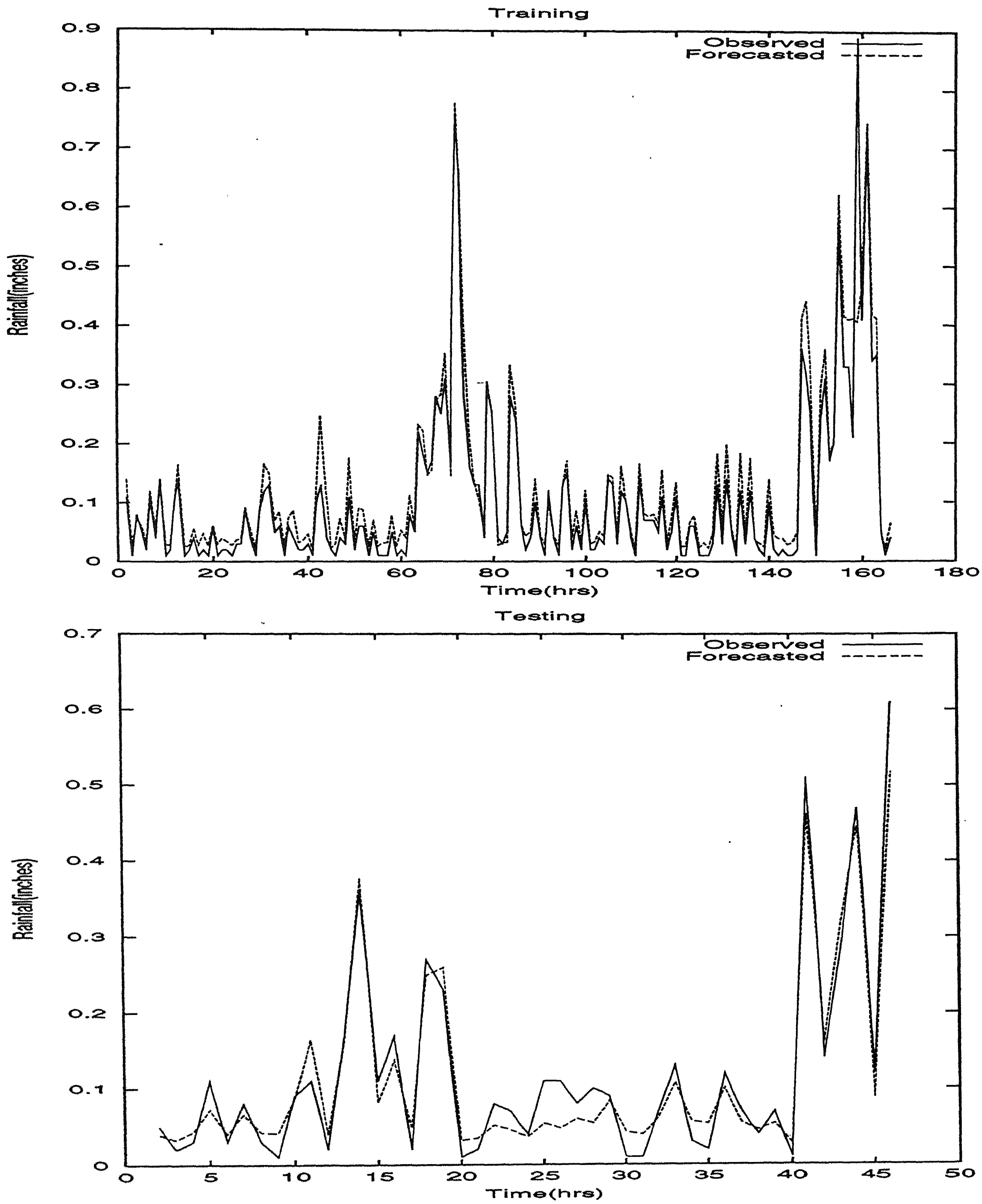


Figure 18: Observed and Forecasted Rainfalls from NLMRM-1 Model

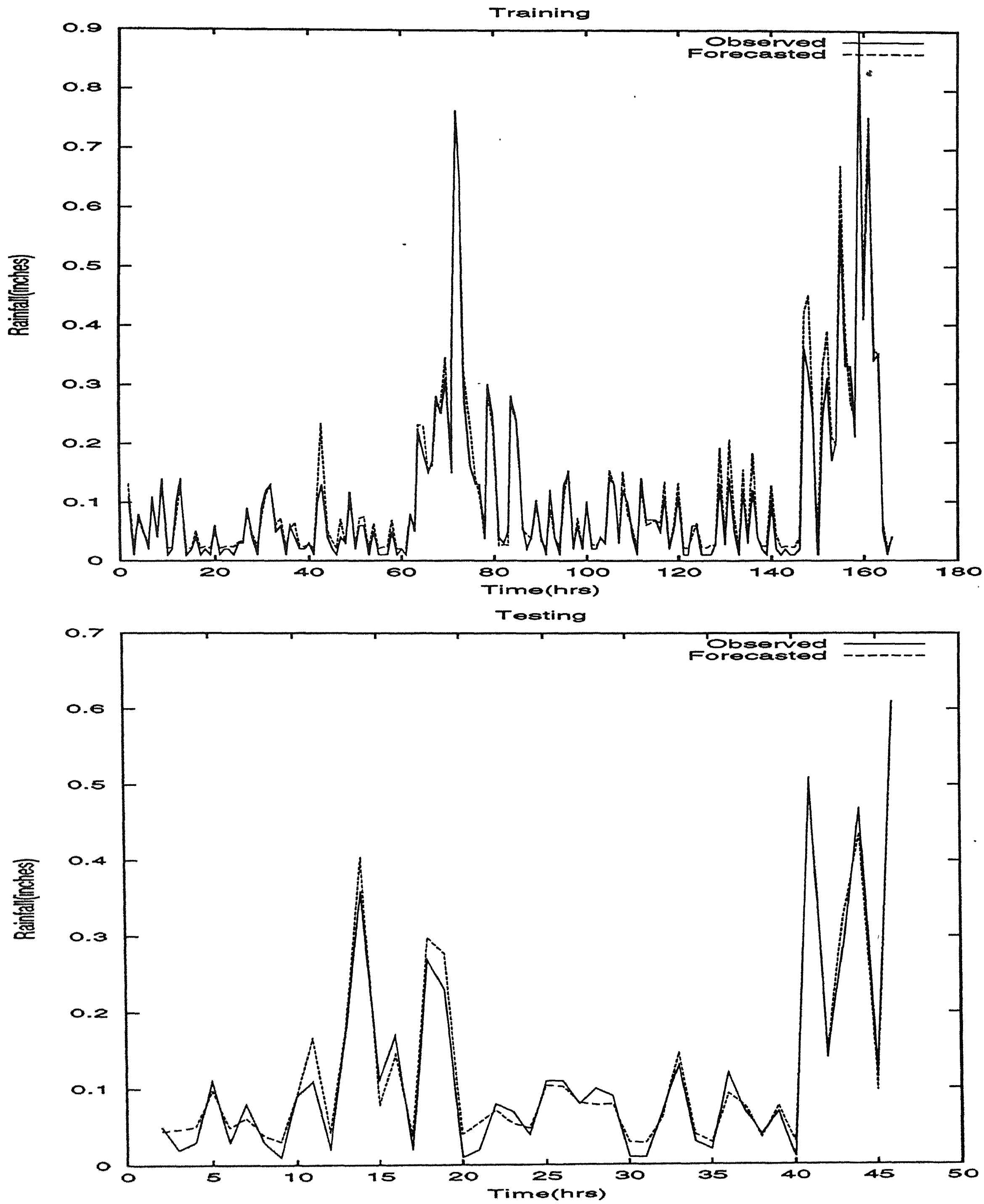


Figure 19: Observed and Forecasted Rainfalls from NLMRM-2 Model

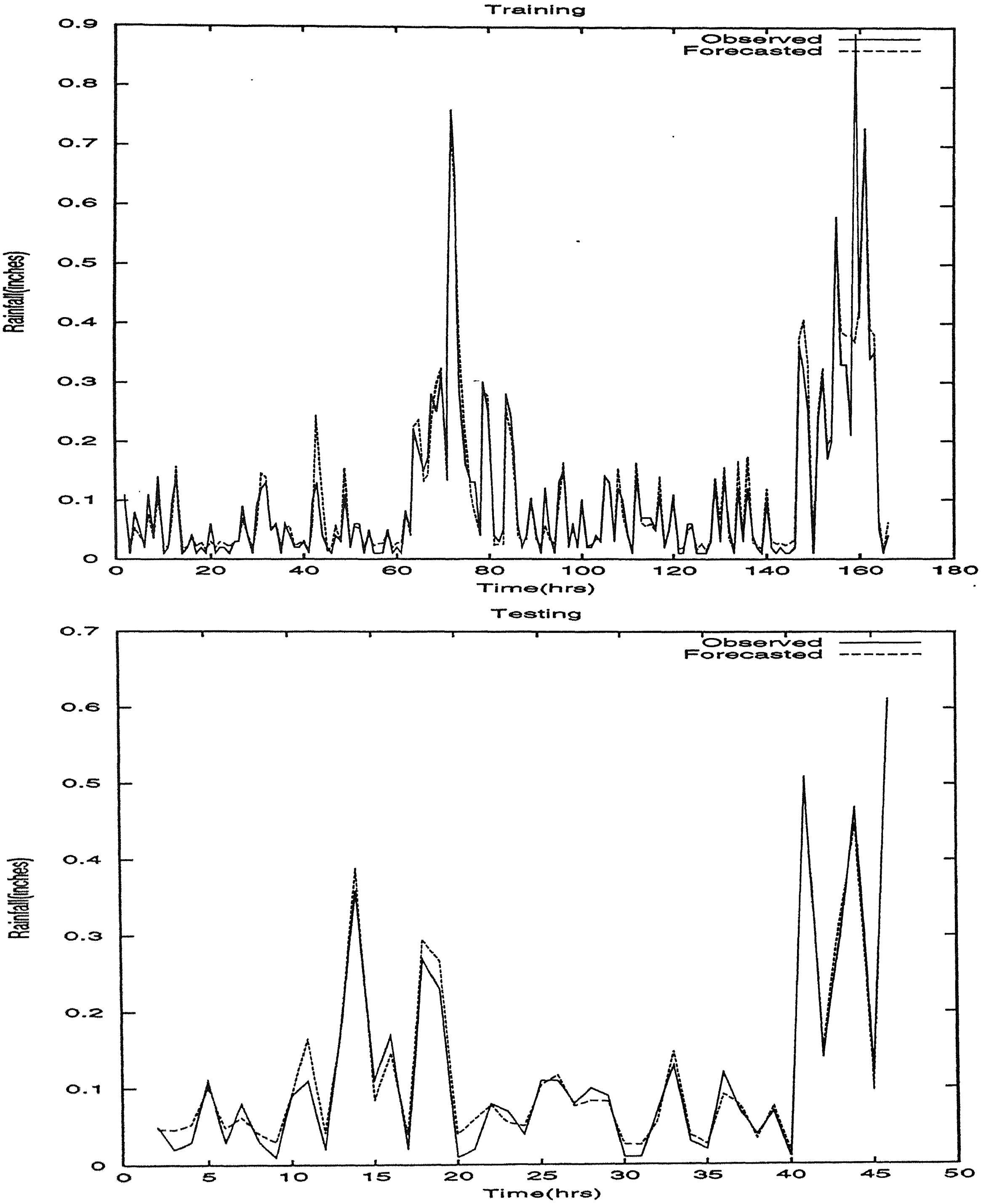


Figure 20: Observed and Forecasted Rainfalls from NLMRM-3 Model

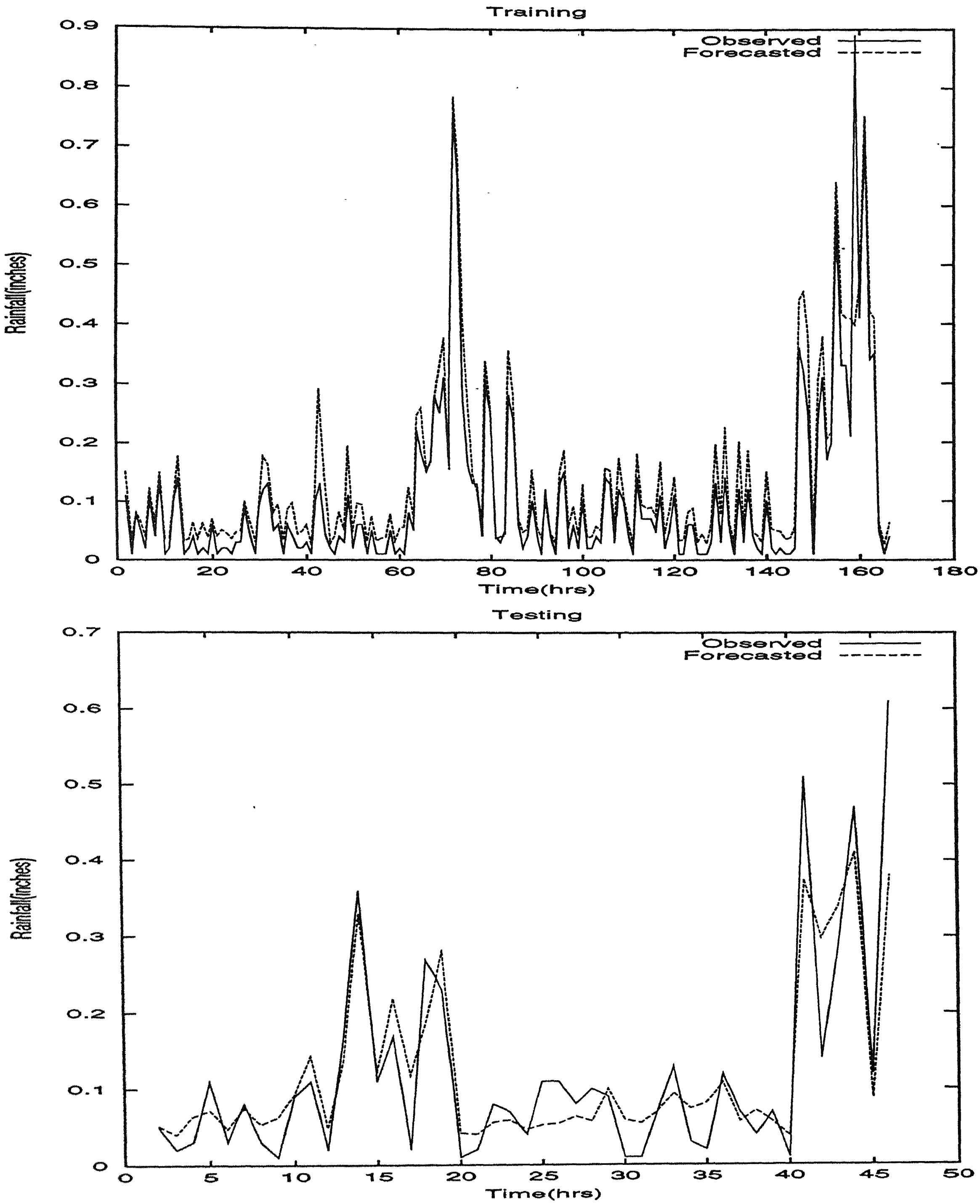


Figure 21: Observed and Forecasted Rainfalls from NLMRM-4 Model

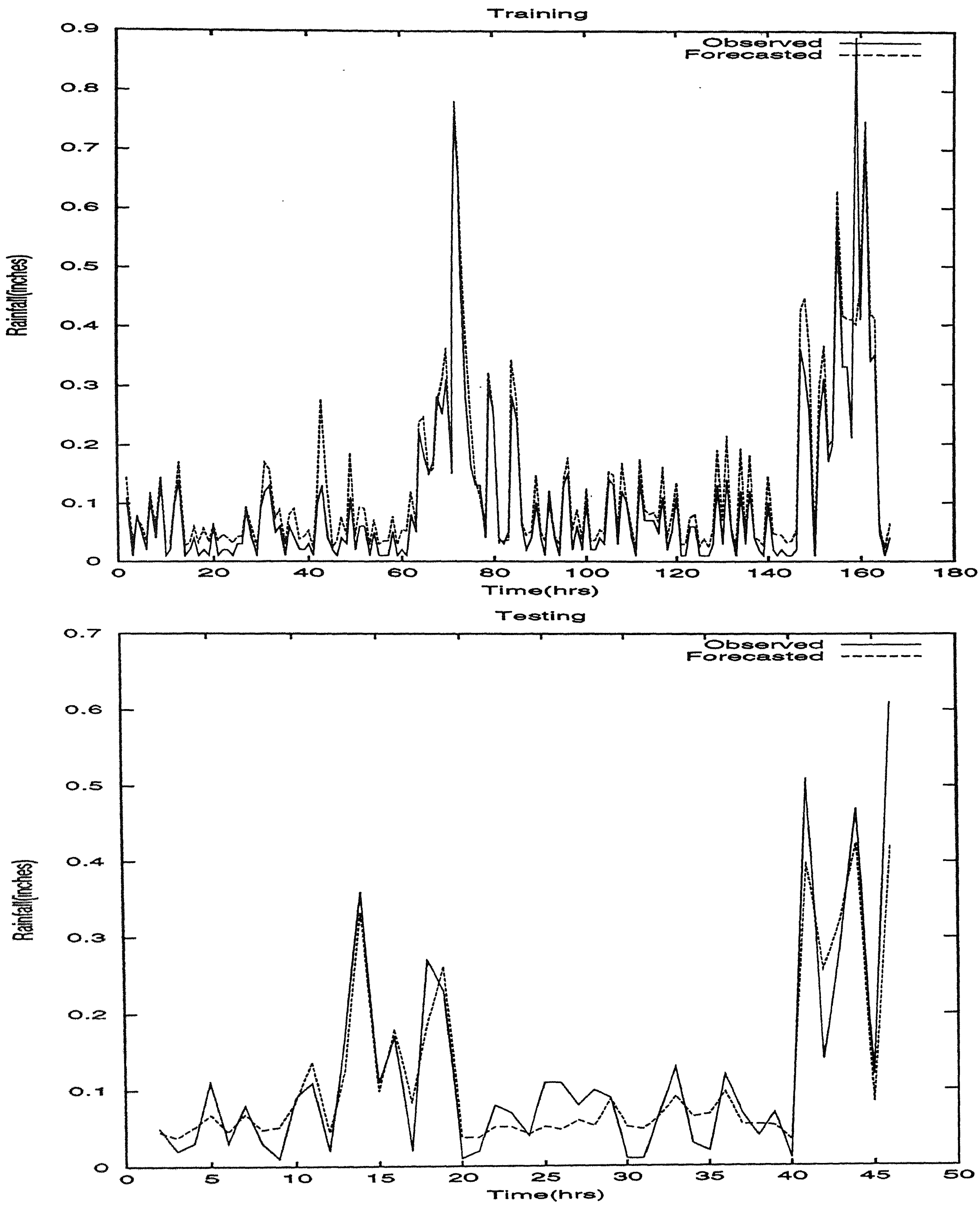


Figure 22: Observed and Forecasted Rainfalls from NLMRM-5 Model

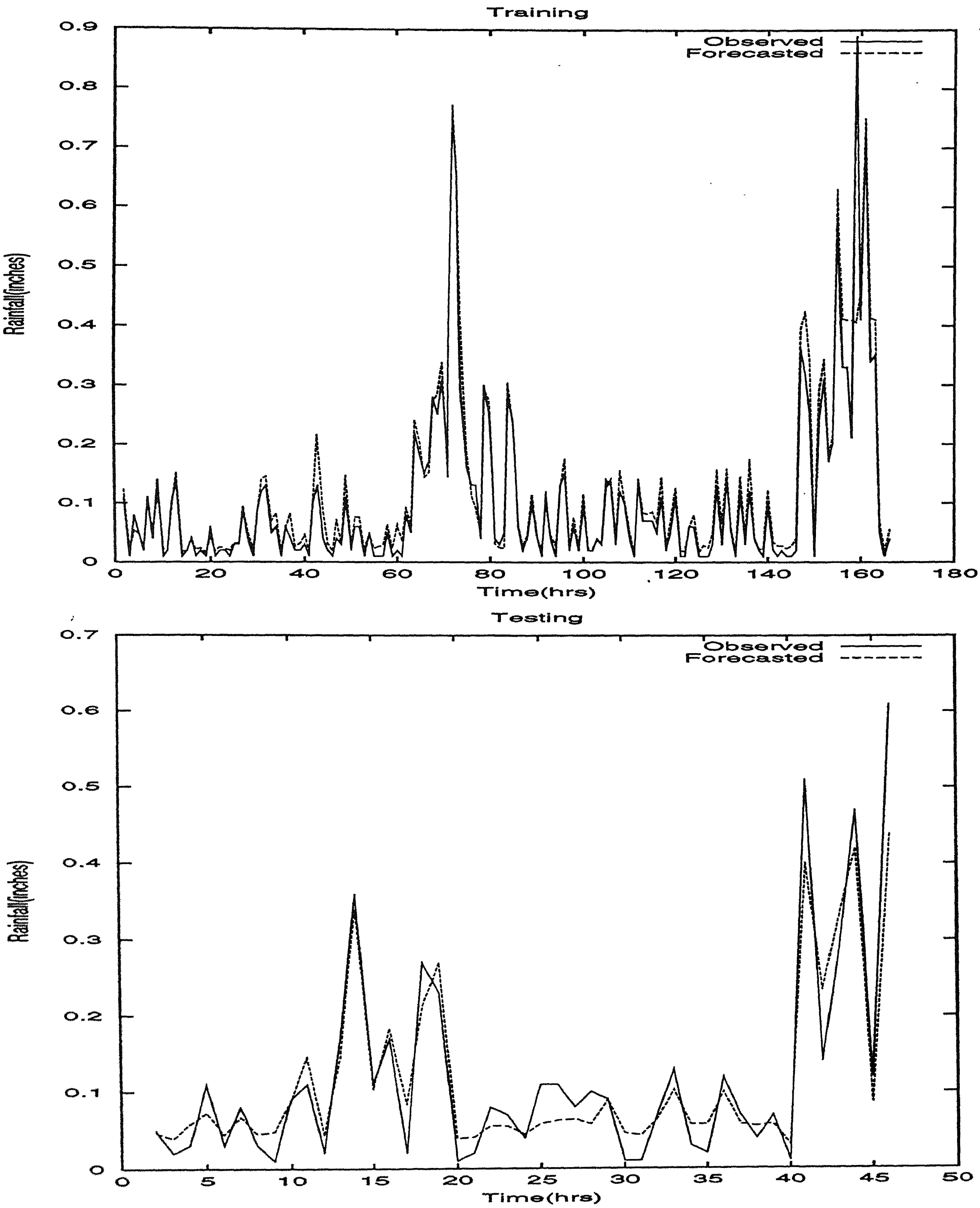


Figure 23: Observed and Forecasted Rainfalls from NLMRM-6 Model

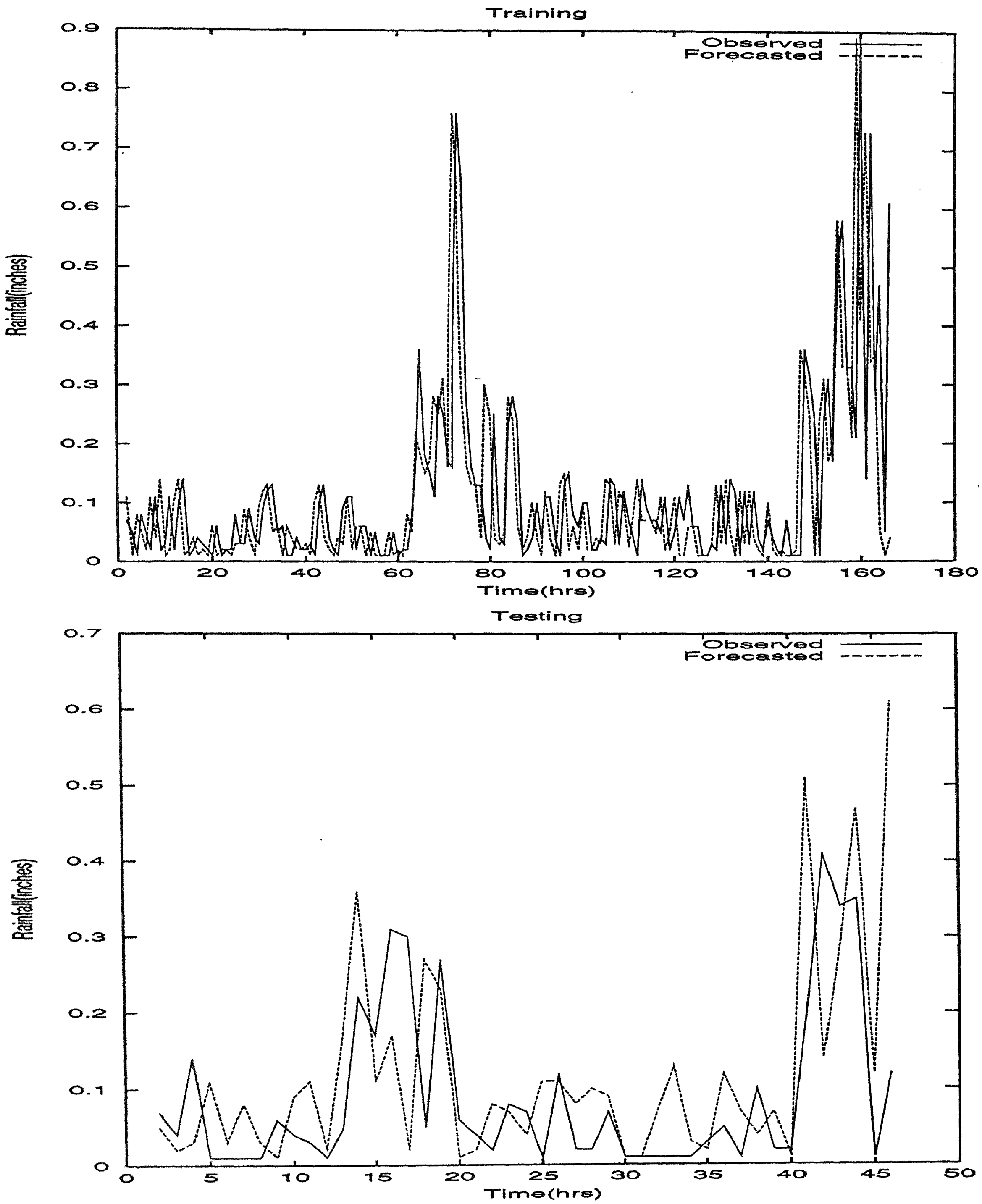


Figure 24: Observed and Forecasted Rainfalls from Persistence Model

Chapter 6

Conclusions

In this study, three types of modeling techniques have been investigated for forecasting hourly rainfall intensities. These are ANN models, Regression models, Thunderstorm Cell models. In addition, a Persistence approach was also utilized, indicating no modeling. In first type of modeling i.e. of ANN modeling two different layered (three and four layers) architectures with 7, 11 and 15 operational variables were explored, same variables were also used in linear and nonlinear regression models. The two other models namely Thunderstorm cell model and Persistence model were also developed. The performance of various structures was evaluated based on various standard statistical parameters. The ANN models have proved to be best among all the models developed in this study.

Accurate quantitative precipitation forecasting for a target area is quite a challenge, especially in a highly convective environment. For one to gain and improve skill in this area, it takes time and various forms of guidance tools. Although only one year of data were available, the results achieved are believed to be reliable and consistent enough to be used for forecasting guidance. Since this was the original goal of the project, the use of Artificial neural networks to predict local rainfall may be expanded to locations in other parts of the country.

The overall conclusion which emerges from this study is twofold. First, the integration of observations with physically based available models does offer some potential for improving rainfall predictions. Second, within the confinements of the current ANN model, the potential gains appear to be small. This conclusion must be viewed as tentative, given the limited testing of the models. It nonetheless corroborates the general notion that progress in short-term rainfall prediction will be laborious and slow. Besides there is a great need for thorough verification studies on large and independent data sets, using comprehensive performance measures.

While it is obvious that limited data are not enough to thoroughly test and train neural networks that deal with ever-changing meteorological variables, it is obviously a good beginning. The verification results show that the networks are reliable and consistent enough to be used as forecasting guidance for Indianapolis Eaglecreek airport. It is possible to improve the networks as more data becomes available, and to develop new networks for other locations as well.

In this study back-propagation algorithm was used for training all the ANN models developed. The performance of ANN models for use in short-term rainfall forecasting may be enhanced by exploring other training algorithm such as unsupervised learning, radial basis function, fuzzy logic and genetic algorithm etc. This is an area which needs further research.

Appendix A

Rainfall Data

Temparature (fahrenhieght)	Dew Point Temp. (fahrenhieght)	Relative Humidity (percentage)	Rainfall (inches)
39	37	92	0.07
47	45	83	0.11
48	46	93	0.07
52	51	96	0.05
52	51	98	0.01
46	44	92	0.08
45	44	94	0.05
39	37	97	0.02
46	43	86	0.11
43	41	94	0.04
40	38	97	0.02
51	49	84	0.14
47	45	94	0.03
45	44	98	0.01
55	53	87	0.11
55	53	97	0.02
55	53	90	0.10
54	52	83	0.14
49	48	98	0.01

Temparature (fahrenhieght)	Dew Point Temp. (fahrenhieght)	Relative Humidity (percentage)	Rainfall (inches)
51	49	98	0.02
48	47	94	0.04
48	47	98	0.01
48	47	95	0.03
60	51	94	0.02
70	58	97	0.01
58	42	91	0.06
52	48	98	0.01
54	53	96	0.02
57	56	96	0.02
57	56	97	0.01
57	57	90	0.08
57	57	95	0.03
57	57	95	0.03
57	56	90	0.09
58	57	93	0.04
50	49	98	0.01
60	59	96	0.03
60	58	90	0.09
53	51	83	0.12
52	51	84	0.13
52	51	92	0.05
52	51	91	0.06
52	51	98	0.01
59	57	92	0.06
62	59	97	0.01
63	60	90	0.04
63	59	95	0.02
65	64	95	0.02
70	68	93	0.03

Temparature (fahrenhieght)	Dew Point Temp. (fahrenhieght)	Relative Humidity (percentage)	Rainfall (inches)
68	65	98	0.01
69	68	87	0.10
71	70	80	0.13
72	69	91	0.04
54	45	96	0.02
56	55	98	0.01
54	54	92	0.04
54	53	86	0.09
55	54	95	0.03
61	61	80	0.11
50	44	80	0.11
47	43	96	0.02
46	44	91	0.06
45	43	91	0.06
44	42	99	0.01
42	40	93	0.05
41	39	99	0.01
40	38	98	0.01
40	38	98	0.01
40	38	92	0.05
49	29	97	0.01
50	28	93	0.02
50	49	98	0.01
52	51	96	0.02
64	57	88	0.08
54	50	92	0.05
58	58	80	0.17
58	58	78	0.22
57	54	70	0.36
57	57	83	0.18

Temparature (fahrenhieght)	Dew Point Temp. (fahrenhieght)	Relative Humidity (percentage)	Rainfall (inches)
57	57	85	0.15
59	58	84	0.17
59	58	88	0.11
60	59	76	0.28
61	60	77	0.25
61	60	73	0.31
58	56	83	0.17
58	57	85	0.16
69	65	63	0.76
67	65	64	0.65
78	76	74	0.28
67	65	85	0.16
69	68	88	0.13
71	70	89	0.13
72	69	92	0.04
71	70	74	0.30
54	45	94	0.02
61	60	75	0.25
59	58	96	0.04
58	57	96	0.03
58	57	96	0.05
57	56	75	0.27
59	58	76	0.23
45	42	74	0.28
44	43	76	0.24
45	43	93	0.06
44	42	98	0.01
37	36	96	0.02
39	37	95	0.04
41	40	96	0.02

Temparature (fahrenhieght)	Dew Point Temp. (fahrenhieght)	Relative Humidity (percentage)	Rainfall (inches)
52	51	91	0.08
46	44	93	0.07
45	44	96	0.04
45	44	86	0.10
45	43	95	0.04
46	44	98	0.01
38	37	89	0.11
38	36	88	0.12
36	35	90	0.11
35	34	96	0.04
35	33	99	0.01
39	30	78	0.13
39	30	70	0.15
53	51	98	0.02
47	44	89	0.08
45	42	91	0.06
40	38	97	0.02
53	50	90	0.10
56	55	88	0.10
51	47	96	0.02
44	42	97	0.02
43	41	95	0.04
42	41	96	0.03
57	52	83	0.14
55	53	85	0.13
57	54	94	0.03
59	57	83	0.12
59	58	89	0.10
57	55	93	0.04
53	49	99	0.01

Temparature (fahrenhieght)	Dew Point Temp. (fahrenhieght)	Relative Humidity (percentage)	Rainfall (inches)
54	52	83	0.14
57	57	91	0.07
57	56	87	0.09
57	56	91	0.07
57	57	91	0.07
57	56	92	0.05
58	53	83	0.11
64	59	96	0.02
62	59	91	0.05
59	58	86	0.11
63	60	97	0.01
62	60	97	0.01
59	58	98	0.01
53	50	90	0.07
52	51	98	0.01
52	52	83	0.13
59	57	92	0.06
59	58	91	0.06
58	57	98	0.01
73	70	96	0.01
60	59	97	0.01
66	63	93	0.03
70	68	93	0.03
68	65	95	0.02
69	68	81	0.13
71	69	93	0.03
72	69	80	0.14
58	57	91	0.05
56	55	85	0.12
54	54	99	0.01

Temparature (fahrenhieght)	Dew Point Temp. (fahrenhieght)	Relative Humidity (percentage)	Rainfall (inches)
54	54	83	0.12
55	54	94	0.03
62	61	82	0.12
57	56	94	0.04
58	56	96	0.02
47	45	98	0.01
48	46	90	0.07
59	54	85	0.10
58	54	95	0.04
50	44	94	0.02
49	45	96	0.01
47	44	96	0.02
46	44	90	0.07
40	39	98	0.01
40	39	98	0.01
41	39	95	0.02
49	28	98	0.01
59	58	66	0.36
59	58	68	0.32
58	57	76	0.25
53	49	99	0.01
57	54	75	0.24
57	56	72	0.31
57	56	83	0.17
59	59	80	0.20
60	59	66	0.51
58	57	67	0.58
64	63	71	0.33
65	63	73	0.33
73	68	78	0.21

Temparature (fahrenhieght)	Dew Point Temp. (fahrenhieght)	Relative Humidity (percentage)	Rainfall (inches)
73	69	87	0.89
71	69	73	0.41
72	69	84	0.14
65	63	64	0.73
66	64	70	0.34
52	47	73	0.29
70	69	72	0.35
62	61	69	0.47
61	60	94	0.05
56	55	99	0.01
56	55	86	0.12
53	53	60	0.61
57	56	96	0.04

Source of Data

The meteorological data at Indianapolis Eaglecreek airport for the year 1997 were downloaded from the internet (*http://shadow.agry.purdue.edu*).

Appendix B

Coke Rate Data of Visakhapatnam Steel Plant

Hot Blast Temp. *(celcius)*	Blast Pres. *(atm.)*	Coke Rate *(kg/m.tonne hour)*
1050	3.90	517
1050	3.00	506
1050	3.20	536
1050	3.20	550
1050	3.09	520
1050	3.20	508
1050	3.20	501
1050	3.20	555
1050	3.20	513
1050	3.20	501
1050	3.29	549
1050	3.29	552
1050	3.29	527
1050	3.20	483
1050	3.20	576
1052	3.20	554
1045	3.20	500

Hot Blast Temp.	Blast Pres.	Coke Rate
celcius	*atm.*	*kg/m.tonne hour*
1045	3.09	532
1030	3.09	509
1040	3.09	529
1030	3.20	547
1015	3.20	593
1005	3.20	491
1025	3.09	526
1035	2.95	482
1040	3.09	542
1030	3.09	534
1030	3.09	532
1040	3.29	500
1030	3.20	509
1040	3.25	547
1030	3.20	525
1035	2.70	583
1045	2.75	530
1025	3.29	572
1038	3.04	523
1045	3.00	478
1048	3.09	646
1030	3.09	479
1030	3.04	494
1014	2.93	547
1011	2.59	533
1023	2.70	522
1026	2.71	538
1040	3.05	528
1030	2.24	507
1032	2.07	534

Hot Blast Temp.	Blast Pres.	Coke Rate
celcius	*atm.*	*kg/m.tonne hour*
1039	2.94	488
1045	2.67	498
1049	3.20	559
1038	2.79	533
1039	2.96	537
1043	2.39	527
1040	2.56	538
1038	3.22	547
1045	2.86	534
1024	2.70	494
1029	3.09	533
1039	2.93	544
1027	2.77	547
1050	3.01	572
1032	2.91	533
1030	3.05	479
1050	3.09	528
1054	3.13	466
1057	3.00	572
1050	2.95	510
1034	3.09	595
1045	3.29	572
1053	2.77	534
1050	3.09	528
1050	2.82	530
1050	2.43	534
1046	2.86	527
1033	2.97	566
1033	2.84	494
1036	2.89	523

Hot Blast Temp.	Blast Pres.	Coke Rate
celcius	*atm.*	*kg/m.tonne hour*
1032	3.02	511
1039	2.78	516
1037	3.01	577
1035	2.97	559
1020	2.85	477
1025	3.11	483
1028	2.76	496
1010	2.63	545
1020	2.93	535
1006	2.48	488
1005	2.30	500
986	2.20	617
1025	2.81	496
1032	2.63	543
1046	2.65	520
1049	3.14	506
1047	2.63	548
1048	1.81	520
1154	2.80	526
1019	2.69	554
1028	2.68	506
1029	2.67	557
1033	2.69	519
1004	2.63	561
999	2.81	594
1011	2.77	515
1018	2.89	527
1014	2.65	553
1016	2.64	611
1023	2.79	547

Hot Blast Temp.	Blast Pres.	Coke Rate
celcius	*atm.*	*kg/m.tonne hour*
1037	3.98	557
1040	3.09	521
1043	3.00	620
986	3.00	536
960	2.96	548
956	2.90	562
1052	3.20	554
1045	3.20	500
1045	3.09	532
1030	3.09	509
1040	3.09	529
1030	3.20	547
1015	3.20	593
1005	3.20	491
1025	3.09	526
1016	2.64	611

Source of Data

Daily operational data for Blast furnace at Visakhapatnam Steel Plant, Visakh atnam, India (Punyasheel, M.Tech Thesis, (1999),"*Prediction of Coke Quality and Coke Rate using Fuzzy Logic and Genetic Algorithms.*

MODEL	AARE	CORR.COEFF	STD. DEV.	TS1	TS5	TS10	TS25	TS50	TS75	TS100
2-5-3-1(ANN)	4.7621	0.9977	0.0104	18.0189	40.6604	70.2370	90.1887	100.0000	100.0000	100.0000

Table 8: Statistical Parameters of ANN Model for Coke data of Steel Plant

Appendix C

Statistical Parameters of Various ANN models

MODEL	AARE	CORR.COEFF	STD. DEV.	TS1	TS5	TS10	TS25	TS50	TS75	TS100
7-3-1	108.5488	0.9224	0.0648	0.4717	3.7736	8.0189	23.1132	41.0377	58.4906	67.4528
7-4-1	70.8853	0.9227	0.0594	2.3585	10.3774	18.3962	35.3774	51.8868	66.0377	73.5849
7-5-1	43.7531	0.9891	0.0233	3.7736	14.1509	23.5849	50.0000	77.3585	83.0189	85.3774
7-6-1	49.9243	0.9830	0.0288	2.8302	15.5660	24.5283	45.2830	70.2830	78.3019	84.4340
7-7-1	46.3142	0.9935	0.0184	6.1321	16.0377	30.6604	52.8302	69.3396	77.8302	84.4340
7-8-1	53.5482	0.9882	0.0250	2.3585	13.6792	24.0566	47.6415	67.9245	73.5849	81.6038
7-9-1	108.3046	0.9758	0.0417	2.3585	8.0189	14.1509	27.3585	44.3396	57.5472	63.2075
7-10-1	53.3440	0.9884	0.0255	2.3585	14.1509	23.1132	42.4528	66.5094	75.9434	82.5472
7-11-1	34.1075	0.9902	0.0211	4.2453	14.6226	28.3019	58.4906	75.4717	87.2642	90.5660
7-12-1	40.9284	0.9908	0.0209	4.7170	16.0377	29.2453	54.7170	73.5849	82.5472	88.2076
7-13-1	19.6342	0.9952	0.0137	6.1321	24.0566	40.0943	73.5849	91.9811	96.6981	99.0566
7-14-1	32.6956	0.9914	0.0192	4.2453	17.4528	29.7170	63.2075	77.3585	88.6792	92.4528
7-15-1	34.0134	0.9921	0.0189	5.1887	19.8113	32.0755	58.4906	75.4717	85.3774	91.9811
7-16-1	27.8728	0.9901	0.0209	8.0189	18.8679	33.4906	67.9245	80.1887	91.0377	96.2264
7-17-1	28.7723	0.9928	0.0175	3.7736	20.7547	36.3208	65.0943	81.1321	91.5094	93.8679
7-18-1	54.9142	0.9895	0.0244	2.8302	11.7925	22.6415	42.9245	64.1509	71.6981	78.7736
7-19-1	28.8647	0.9919	0.01856	2.8302	20.2830	33.9623	63.6792	80.6604	91.0377	96.2264
7-20-1	32.0287	0.9916	0.0192	6.6038	20.7547	30.6604	61.3208	78.3019	89.1509	93.3962
7-21-1	43.1151	0.9907	0.0212	6.1321	17.9245	26.8868	52.8302	72.6415	79.2453	87.7358
7-22-1	29.6174	0.9925	0.01825	2.8302	18.8679	33.4906	62.2642	81.6038	88.6792	94.3396
7-23-1	35.3928	0.9918	0.0191	4.7170	19.8113	31.1321	62.7358	75.9434	83.9623	90.5660
7-24-1	27.4352	0.9929	0.0173	4.2453	20.2830	34.4340	64.6226	82.5472	90.5660	95.7547
7-25-1	35.5799	0.9919	0.0189	2.8302	18.8679	36.3208	60.8491	77.8302	84.9057	92.4528

Table 9: Statistical Parameters of 7-N-1 ANN Models

MODEL	AARE	CORR.COEFF	STD. DEV.	TS1	TS5	TS10	TS25	TS50	TS75	TS100
11-8-1	54.3827	0.9893	0.0241	3.7736	16.9811	28.3019	45.2830	68.3962	78.3019	83.4906
11-9-1	31.8598	0.9944	0.0152	7.5472	24.0566	36.3208	57.0755	81.1321	89.1509	92.4528
11-10-1	22.8634	0.9948	0.0141	5.6604	23.5849	39.6226	66.5094	88.2076	94.8113	97.6415
11-11-1	39.1730	0.9908	0.0207	2.8302	8.0189	17.4528	49.5283	75.4717	85.8491	91.0377
11-12-1	42.4154	0.9939	0.0178	4.2453	15.5660	29.2453	55.1887	72.1698	80.6604	87.7358
11-13-1	34.3270	0.9936	0.0170	2.8302	15.0943	32.5472	57.5472	75.9434	87.2642	93.8679
11-14-1	27.7889	0.9942	0.01555	8.4906	23.5849	37.2642	68.3962	81.6038	90.5660	96.6981
11-15-1	22.0135	0.9953	0.0134	9.9057	27.3585	38.2075	66.5094	90.0943	96.2264	98.5849
11-16-1	26.7666	0.9958	0.0131	7.0755	26.4151	39.6226	65.0943	84.4340	91.5094	94.8113
11-17-1	42.0094	0.9934	0.0186	3.3019	13.2075	27.8302	54.7170	70.2830	82.5472	89.6226
11-18-1	18.1252	0.9966	0.0115	8.0189	26.4151	43.8679	76.8868	93.3962	96.6981	98.5849
11-19-1	19.0764	0.9977	0.0096	8.0189	31.1321	52.3585	76.4151	89.6226	94.8113	98.5849
11-20-1	29.7494	0.9951	0.0145	7.5472	24.5283	35.8491	65.0943	80.6604	88.2076	93.3962
11-21-1	24.4224	0.9957	0.0129	5.1887	24.0566	41.5094	70.7547	86.3208	92.4528	95.7547
11-22-1	25.184123	0.9959	0.0131	7.0755	27.3585	40.0943	68.8679	84.4340	92.4528	96.6981
11-23-1	32.2154	0.9936	0.0170	5.6604	18.8679	33.4906	63.6792	78.7736	86.7924	91.9811
11-24-1	25.9476	0.9956	0.0131	8.4906	23.5849	40.5660	69.8113	83.0189	91.0377	95.2830
11-25-1	23.2137	0.9960	0.0122	6.1321	26.4151	41.5094	68.8679	86.3208	93.3962	98.1132
11-26-1	22.8597	0.9958	0.0129	8.4906	25.4717	41.5094	70.2830	86.7924	93.8679	97.6415
11-27-1	30.2023	0.9945	0.0156	3.3019	13.6792	27.3585	59.9057	83.9623	89.1509	95.2830
11-28-1	18.6351	0.9966	0.0116	11.3208	28.7736	43.8679	72.6415	91.9811	97.6415	99.0566
11-29-1	25.9313	0.9964	0.0124	4.7170	23.5849	38.2075	68.3962	83.0189	92.4528	96.2264
11-30-1	26.6165	0.9953	0.0137	6.6038	23.5849	40.0943	67.9245	83.4906	90.0943	96.2264

Table 10: Statistical Parameters of 11-N-1 ANN Models

MODEL	AARE	CORR.COEFF	STD. DEV.	TS1	TS5	TS10	TS25	TS50	TS75	TS100
15-10-1	19.5163	0.9966	0.0121	10.8491	25.4717	41.0377	68.3962	92.4528	97.6415	99.0566
15-11-1	18.3072	0.9969	0.0117	13.2075	29.7170	46.2264	71.2264	93.3962	97.6415	99.5283
15-12-1	20.8169	0.9966	0.0123	5.6604	25.9434	39.6226	69.8113	91.9811	96.2264	98.1132
15-13-1	21.1565	0.9970	0.0115	9.4340	26.8868	40.0943	68.3962	89.1509	96.2264	98.1132
15-14-1	19.1816	0.9967	0.0118	6.1321	28.3019	44.8113	71.6981	91.9811	96.6981	98.584
15-15-1	19.1458	0.9970	0.0110	12.2642	28.3019	45.7547	73.5849	91.9811	96.6981	98.5849
15-16-1	27.8734	0.9969	0.0130	8.0189	20.7547	34.9057	63.2075	84.4340	89.6226	95.7547
15-17-1	24.0481	0.9969	0.0125	6.6038	21.2264	33.0189	61.7925	89.6226	95.2830	97.1698
15-18-1	19.1442	0.9970	0.0113	12.2642	27.8302	46.2264	72.1698	91.9811	97.1698	99.0566
15-19-1	21.2448	0.9974	0.0110	9.4340	28.3019	42.9245	67.9245	90.5660	94.8113	98.5849
15-20-1	20.2569	0.9970	0.0116	10.8491	23.1132	41.5094	67.9245	90.0943	97.6415	98.1132
15-21-1	21.7215	0.9969	0.0119	10.3774	25.0000	40.0943	66.5094	91.5094	96.2264	98.1132
15-22-1	33.5610	0.9968	0.0150	3.7736	15.5660	25.4717	50.0000	79.7170	89.6226	94.8113
15-23-1	27.6820	0.9968	0.0134	8.0189	17.9245	31.6038	58.4906	87.7358	91.0377	95.7547
15-24-1	22.5511	0.9966	0.0120	8.9623	24.5283	36.3208	70.7547	88.2076	95.7547	97.6415
15-25-1	23.1309	0.9969	0.0115	9.4340	23.5849	40.0943	70.7547	87.7358	93.3962	96.6981
15-26-1	22.3039	0.9969	0.0118	11.3208	24.5283	42.4528	68.8679	88.6792	95.2830	97.6415
15-27-1	22.6513	0.9973	0.0112	9.9057	29.2453	41.0377	69.3396	87.7358	94.3396	96.6981
15-28-1	20.3818	0.9972	0.0110	13.2075	28.7736	44.8113	71.2264	91.5094	96.2264	98.1132
15-29-1	21.9525	0.9971	0.0115	11.3208	25.9434	40.5660	68.8679	90.0943	93.8679	97.6415
15-30-1	29.5567	0.9956	0.0150	5.6604	19.8113	27.8302	61.7925	84.4340	91.0377	94.3396

Table 11: Statistical Parameters of 15-N-1 ANN Models

MODEL	AARE	CORR.COEFF	STD. DEV.	TS1	TS5	TS10	TS25	TS50	TS75	TS100
7-8-5-1	17.0529	0.9968	0.0117	5.6604	28.3019	45.7547	78.3019	94.3396	99.0566	99.5283
7-8-7-1	15.4312	0.9977	0.0100	8.0189	30.6604	50.0000	80.1887	94.3396	100.0000	100.0000
7-9-6-1	14.3041	0.9977	0.0095	9.2233	32.0388	49.0291	85.4369	99.5146	100.0000	100.0000
7-10-5-1	14.4135	0.9977	0.0099	9.9057	30.1887	46.6981	82.5472	97.1698	99.5283	100.0000
7-10-6-1	15.3704	0.9976	0.0102	12.2642	33.0189	45.2830	75.9434	96.6981	99.5283	100.0000
11-11-7-1	28.2087	0.9939	0.0170	5.1887	17.4528	27.3585	56.1321	82.5472	93.3962	96.6981
11-12-6-1	21.9560	0.9961	0.0137	8.4906	25.9434	37.2642	67.9245	90.5660	94.8113	98.1132
11-13-5-1	14.3422	0.9969	0.0119	11.7925	25.0000	44.3396	69.3396	93.3962	99.5283	100.0000
11-13-6-1	18.0063	0.9973	0.0111	10.8491	27.3585	43.3962	68.3962	93.8679	99.5283	100.0000
11-13-7-1	24.9945	0.9937	0.0174	6.6038	13.6792	24.5283	58.0189	88.6792	98.1132	100.0000
15-8-5-1	23.1701	0.9972	0.0115	10.3774	21.6981	37.2642	69.8113	88.2076	94.8113	97.1698
15-8-6-1	27.2205	0.9965	0.0144	6.6038	16.9811	27.3585	61.3208	86.7924	93.8679	96.6981
15-8-7-1	21.7305	0.9968	0.0118	9.4340	28.7736	43.8679	67.9245	87.2642	96.2264	98.5849
15-9-6-1	22.5585	0.9967	0.0120	11.3208	26.4151	41.5094	70.7547	88.2076	93.8679	96.6981
15-9-7-1	24.8010	0.9971	0.0120	7.0755	24.0566	39.6226	66.9811	87.2642	92.4528	96.2264
15-10-5-1	21.8471	0.9974	0.0108	8.4906	26.4151	42.9245	71.6981	88.6792	93.8679	98.1132
15-10-6-1	21.8941	0.9966	0.0122	9.9057	22.6415	39.6226	71.6981	89.1509	95.7547	97.6415
15-10-7-1	22.8327	0.9977	0.0109	9.4340	25.0000	43.8679	68.3962	88.2076	95.2830	96.6981
15-11-5-1	24.5570	0.9975	0.0107	4.2453	24.0566	43.3962	65.0943	86.3208	92.9245	96.2264
15-11-6-1	16.6524	0.9975	0.0108	12.2642	30.1887	43.3962	76.4151	95.7547	98.5849	100.0000
15-11-7-1	14.0126	0.9983	0.0089	11.7925	34.4340	56.6038	80.1887	97.1698	100.0000	100.0000
15-11-8-1	13.4150	0.9984	0.0084	17.4528	39.1509	55.6604	80.1887	97.1698	100.0000	100.0000

Table 12: Statistical Parameters of Four-layer ANN Models

Bibliography

1. Aleksander, I. and Morton, H. (1990), "An Introduction to Neural Computing", Chapman & Hall, London.

2. Chen, Z.Q. and Kavvas, M.L. (1992),"An Automated Method for Representing, Tracking and Forecasting Rain Fields of Severe Storms by Doppler Weather Radars, Journal of Hydrology, No. 132, pp 179-200.

3. Churchland, P.S. and Sejnowski, T.J. (1992),"The Computational Brain", MIT Press, Cambridge.

4. Dimitris, T., Haferman, J.L., Anagnostou, E., Krajewski, W.F. and Smith, T.F. (1997),"Physically Based Satellite Retrieval of Precipitation using a 3D Passive Microwave Radiative Transfer Algorith", Journal of Atmospheric and Oceanic Technology, Vol. 13, pp 832-850.

5. Faggin F.(1991),"VLSI Implementation of Neural Networks", Tutorial Notes,International Joint Conference on Neural Networks, Seattle, W.A.

6. French, M.N. and Krajewski, W.F.(1994), "A Model for Real-Time Quantitative Rainfall Forecasting using Remote Sensing", Water Resources Research, Vol. 30, No. 4, pp 1075-1083.

7. French, M.N., Krajewaski, W.F. and Cuykendall R.R.(1992), "Rainfall forecasting in Space and Time using a Neural Network", Journal of Hydrology, No. 137, pp 1 -31.

8. Georgakakos, K.P. and Krajewski, W.F. (1991), "Worth of Radar data in Real-Time Precipitation of Mean Areal Rainfall by Non-Advective Physically-Based models", Water Resources Research, Vol. 27, No.2, pp 185-197.

9. Glahn, H.R. and Lowry, D.A. (1972),"The Use of Model Output Statistics(MOS) in Objective Weather Forecasting", Journal of Applied Meteorology, Vol. 11, pp 1203-1211.

10. Haykin S.(1994)," Neural Networks - A Comprehensive Foundation, Macmillan College Publishing Company

11. Hershenhorn, J. and Woolhiser, D.A. (1987), " Disaggregation of daily Rainfall", Journal of Hydrology, No. 95, pp 299-228.

12. Hsu, K., Gupta, V.K. and Soroosh S. (1993),"Artificial Neural Network Modeling of the Rainfall-Runoff Process", Water Resources Research, Vol. 29, No. 4, pp 1185-1194.

13. McCullagh, J.(1997),"The Estimation of Rainfall by Artificial Neural Networks", Seminar at E Day (Environmental Day), LaTrobe University, Bendigo, Australia.

14. Moriyama, T. and Muneo, H. (1994), "Quantitative Precipitation Forecasting using Neural Networks",*unesco.org.uy/phi/libros/radar/art24.html.*

15. National Weather Service, Meterological Service Division(NWS-MDS) (1981), " Two to Six hour Probabilities of Thunderstorm and Severe Local storms", National Weather Service, National Oceanic and Atmospheric Administration, Silver Spring, Maryland, Teechnical Procedures Bulletin, S.No. 295, pp 13.

16. Ozelkan, C.E., Ni, F. and Duckstein, L. (1996), " Relationship between Monthly Atmospheric Circulation Patterns and Precipitation: Fuzzy Logic and Regression approaches", Water Resources Research, Vol. 32, No. 7, pp 2097-2103.

17. Park *et al.* (1991),"Electrical Load Forecasting using an Artificial Neural Network", IEEE Transactions on Power Systems, Vol.6, No. 12, pp 442-449.

18. Ramon Y. Cajal S. (1911),"Histologie du systeme nervous de l'homme *et* des vertebres, English translation, by N. and L.W. Swanson (1994), Oxford University Press.

19. Rumelhart, D.E. and McClelland, J.E. (1986),"Parallel Distributed processing", Vol. 1, UHI, MIT Press, Cambridge.

20. Scardi, M. (1998), "Empirical Models in Ecology: Some Applications of Artificial Neural Networks",VII International Congress of Ecology, Florence,Symposium 5.6, 21 July 1998.

21. Seo, D.J. and Smith, J.A. (1992), "Radar-based Short-Term Rainfall Prediction", Journal of Hydrology, No. 131, pp 341-367.

22. Sharda, J.D. and Patil, R.B. (1990), "Connectionist Approach to Time Series Prediction: An Empirical Test", Technical Report No. 90-26; 11-90, College of Business Administration, Oklahoma State University, Oklahoma.

23. Tommy W.S. Chow and S.Y. Cho (1997), "A Novel Neural Based Rainfall Nowcasting System in Hong Kong", City University of Hong Kong, Kowloon.

24. Zadeh, L.A. (1965), "Probability Measures of Fuzzy Events", Journal of Mathematical Analysis Application, Vol. 23, pp 421-437.

25. Zhiquan, B., Shafiqul, I. and Elathir, E.A.B. (1994), "Aggregation- Disaggregation Properties of a Stochastic Rainfall Model", Water Resources Research, Vol. 30, pp 3423-3435.

26. Zurada, J.M. (1997),"Introduction to Artificial Neural Systems, Jaico Publishing House, Delhi.